FINAL REPORT

ON

NASA PROJECT NAG 3-1376

DEVELOPMENT OF A MODEL BASED TECHNIQUE FOR GEAR DIAGNOSTICS USING THE WIGNER-VILLE METHOD

by
F. Choy
A. Xu
V. Polyshchuk
Department of Mechanical Engineering
The University of Akron
Akron, OH 44325-3903

ABSTRACT

Imperfections in gear tooth geometry often results from errors in the manufacturing process or excessive material wear during operation. Such faults in the gear tooth geometry can result in large vibrations in the transmission system, and, in some cases, may lead to early failure of the gear transmission system. This report presents the study of the effects of imperfection in gear tooth geometry on the dynamic characteristics of a gear transmission system. The faults in the gear tooth geometry are modeled numerically as the deviation of the tooth profile from its original involute geometry. The changes in gear mesh stiffness due to various profile and pattern variations are evaluated numerically. The resulting changes in the mesh stiffness are incorporated into a computer code to simulate the dynamics of the gear transmission system. A parametric study is performed to examine the sensitivity of gear tooth geometry imperfections on the vibration of a gear transmission system. The parameters varies in this study aconsist of the magnitude of the inperfection, the pattern of the profile variation, and the total number of teeth affected. Numerical results from the dynamic simulations are examined in both the time and the frequency domains. A joint time-frequency analysis procedure using the Wigner-Ville Distribution is also introduced to identify the location of the damaged tooth from the vibration signature. Numerical simulations of the system dyanmics with gear faults were compared to experimental results. An optimal tracker was introduced to quantify the level of damage in the gear mesh system. Conclusions are drawn from the results of this numerical study.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Gearing, one of the most universally used machine elements, is applied in mechanical systems of every size and description: from the tiny pinions in a watch or a computer system to the high-speed, heavily-loaded reduction gears of an aircraft gas turbine. In the last two decades, the use of gear transmissions in both defense and commercial applications has substantially increased. With the demand for higher power and performance, premature failures in transmissions often result in financial losses, and sometimes even lead to catastrophic consequences. In the aerospace industry, where both weight-to-load factor and efficiency are pushed to their design limits, one of the major concerns is fatigue failures in rotorcraft gear transmission systems. Such failures are often a result of excessive gear tooth wear and tooth crack formation. Presently, the prevention and management of premature equipment failures has become a vital part of the maintenance program.

Current on-board condition monitoring systems for gas-turbine engine systems often fail to provide sufficient time between warning and failure such that safety procedures can be implemented. On the other hand, inaccurate interpretation of operational conditions may result in false alarms and unnecessary repairs and downtime. The early detection of incipient failure in a mechanical system is of great practical importance as it permits scheduled inspections without costly shutdowns, along with indication of urgency and location for repair before any catastrophic failure.

Presently, a considerable amount of work in machine life prediction has been carried out using machine reliability and design life approaches[1,2]. However, most of this work is based on statistical predictions developed by Lundberg and Palmgren[3] without considering the conditions of machine components during various phases of their lifespan. Besides the work reported in recent years by the Mechanical Failure Prevention Group(MFPG)[4 - 6], very little has been cited in the literature concerning condition-based failure prediction.

The increasing requirements for long life and safe operation in mechanical systems call for the development of an accurate machine fault identification and failure prognostication system which is capable of on-line machine health monitoring as well as machine life prediction. One of the advanced fault identification procedures used in rotorcraft mechanical systems is the signature analysis of machine vibration/acoustic signals[7 - 9]. The acquired machine vibration/acoustic signature is compared to a data bank of standard healthy machine operation signatures to pinpoint the abnormalities of the input signal. This procedure does not require a shut down of the rotating machinery, and can be used as an in-flight diagnostic and trend monitoring device.

In order to develop an accurate machine fault identification and failure prognostication system, it is very important to understand the dynamics of a gear transmission system under a variety of fault conditions, as well as under nominal conditions. This study deals only with the basic knowledge in fault identification of gear component and the dynamic modelling of a gear transmission system under the effects of gear tooth geometry imperfection.

The major objective of the research presented herein is the development of a numerical model to predict the vibration in a gear transmission system due to gear tooth imperfection or damage. The imperfection/damage of the gear tooth geometry is modeled numerically as the deviation of tooth profile from its original geometry. To simulate gear failures, a computer code developed at NASA Lewis Research Center[10] was modified to simulate various types of gear mesh conditions. The changes in mesh stiffness due to the effects of tooth wear can be represented by using a tooth profile modification procedure. The resulting changes in mesh stiffness are incorporated into the dynamic simulation of the gear transmission system.

In simulating the vibration of the transmission system, the equations of motion are established individually for each rotor-gear-bearing system. These localized changes in the gear mesh stiffness are incorporated into each gear-rotor model during the dynamic simulation [11-13]. The dynamics of each gear-rotor system are coupled with each other through the gear mesh interacting forces and the bearing supporting forces. The global vibrations of the system are evaluated by solving the set of coupled transient dynamics

equations of all the rotor systems simultaneously including the vibration of the casing. In order to minimize the computational effort, the number of degrees-of-freedom of the system are reduced by using a modal synthesis procedure[11,12].

Results from the model were examined using a joint time-frequency analysis method. This approach was chosen because the joint time-frequency analysis will provide an instantaneous frequency spectrum of the system at every instant of the revolution of the pinion while the traditional frequency analysis can only provide the average vibration spectrum of the signal. In other words, the time-changing spectral density from the joint time-frequency spectra will provide vital information concerning the frequency distribution concentrated at a particular instant. The Wigner-Ville Distribution (WVD) [14-16] was chosen for the joint-time frequency analysis. Considerable success has been achieved in applying the WVD to gear transmission systems [17,18] to recognize faults at various locations of the gear.

In addition, experimental results obtained from a gear failure test rig at NASA Lewis Research Center were also used to experimentally validated the identification procedure using the WVD as well as to verify the numerical simulations. A technique for quantifying the damage in a gear mesh using an optimal tracker was also developed, and results obtained using the optimal tracker were compared with those from the experimental studies.

Based on results of this study, general conclusions are made concerning the effects of local damage on the global damage of the gear transmission system. This study applied the above discussed methodology on a variety of damaged gear models. The numerical model used to simulate the dynamics of a gear transmission system with gear tooth geometry imperfections was successfully developed. In addition, considerable success was achieved in generating a comprehensive database of the vibration signal due to various kinds of gear tooth geometry imperfection patterns a in gear transmission system. Some limited success was achieved in quantifying the damage using an optimal tracker. Using the developed analytical/numerical model, a gear more extensive fault/damage database can be developed for machine fault identification and failure prognostication research.

CHAPTER 2

THE EFFECTS OF GEAR TOOTH IMPERFECTION ON GEAR MESH STIFFNESS

2.1. Objective

The magnitude variation in the tooth mesh stiffness can affect gear mesh dynamics and loading significantly. The first step in determining the effects of gear tooth imperfections on mech dynamics is to determine the relationship between gear tooth imperfections and the resulting change to gear mesh stiffness.

The objective of this chapter is to develop a relationship describing the effects of gear tooth imperfections on the static behavior of a gear system, with an emphasis on the gear mesh stiffness. The method which calculates the tooth mesh stiffness of a gear system is presented by Richardson [19]. The imperfection of a gear tooth is modeled numerically as the deviation of the tooth profile from its designed geometry. The changes in gear mesh stiffness due to various profile changes and imperfection patterns are evaluated numerically. A computer code developed at NASA Lewis Research Center[10] was modified to simulate various types of gear mesh conditions.

2.2. Gear Kinematic Properties

The ideal kinematic requirement for gear action is *constant speed ratio*. That is, the angular velocity of the driven gear should be a constant multiple of the angular velocity of the driving gear. Two curves that possess the property of constant speed ratio when operated as contacting tooth surfaces are called conjugate curves. From the infinite

variety of possible conjugate curves, the involute has been almost universally accepted for use in gearing. An involute curve is generated by the end of a line that is unwound from the circumference of a circle called the base circle.

Figure 1 shows two gear teeth in contact. Point L on gear 2 is in contact with point M on gear 1. At this point of contact, the two tooth surfaces must be tangent to each other and consequently must have a common normal W_1W_2 passing through the point of contact which intersects the line of centers C_1C_2 at the instant center P, called the *pitch point*. Since ideal gears are assumed rigid, the velocities of L and M along the normal W_1W_2 must be equal. The velocities of L and M perpendicular to the normal sliding or relative velocity of the tooth surfaces.

$$V_{t2} = w_2 \overline{C_2W}_2 ; \qquad V_{t1} = w_1 \overline{C_1W}_1 \tag{2.1}$$

where w_1 and w_2 are the angular velocities of gears 1 and 2, respectively. Hence, by similar triangles

$$\frac{w_1}{w_2} = \frac{R_2}{R_1} \tag{2.2}$$

This equation is frequently used to define the law of gearing, which states that the pitch point must remain fixed on the line of centers. This means that all the lines of action for every instantaneous point of contact must pass through the pitch point. Consequently, ideal gears can be represented kinematically by two imaginary cylinders of radii R_1 and R_2, called *pitch cylinders, which roll on each other without slipping.*

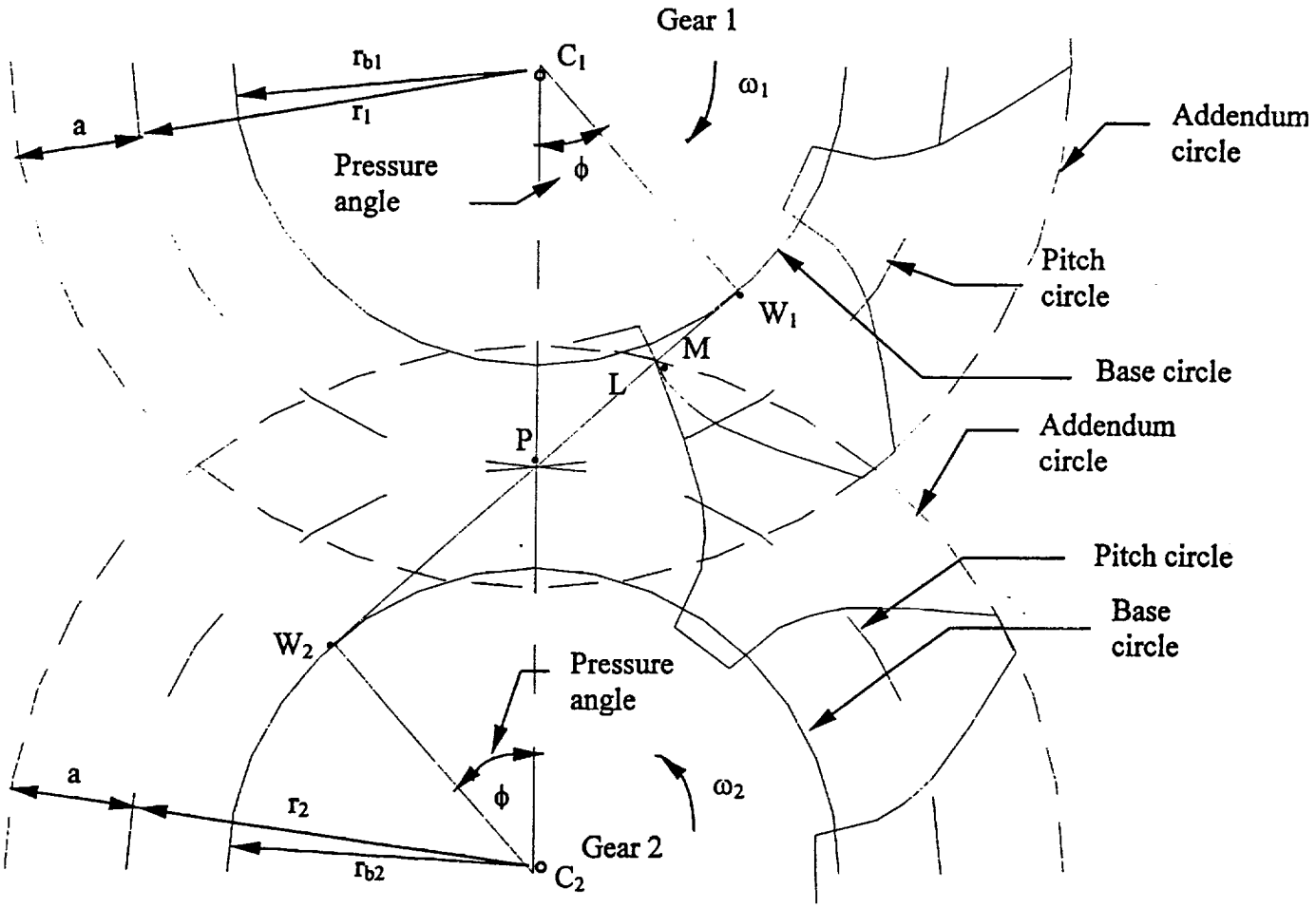


Figure 1. Involute gear geometry

If no friction is present between the mating gear profiles, then the resultant force transmitted at the contact point L must lie along the common normal. For this reason the common normal is called the *pressure line*, and the angle between the normal and a line perpendicular to the line of centers C_1C_2 is called the pressure angle θ. The locus formed by all points of contact as the gears rotate is known as the *path of contact*.

In order to maintain continuous conjugate action, a series of conjugate curves are spaced uniformly around the circumference of a gear. The separation of these curves, measured along the pitch circle, is called the *circular pitch*

$$P_c = \frac{2\pi R}{i} \tag{2.3}$$

where i is the number of teeth, and R is the pitch-circle radius. The *diametrical pitch* is defined as the number of teeth on the gear per inch of pitch diameter, as indicated in the following equation;

$$P = \frac{i}{2R} \tag{2.4}$$

The relationship between circular pitch and diametral pitch is as follows:

$$P_c P = \pi \tag{2.5}$$

In an involute gear, the spacing of successive involutes along the pressure line or line of action is known as the normal pitch, and is related to the circular pitch defined by Equation (2.3) in the following way

$$P_n = P_c \cos\theta \tag{2.6}$$

The *contact-ratio* is defined as the path of contact divided by the normal pitch, and is a measure of the average number of tooth-pairs in contact. To provide continuous action the contact ratio must be greater that one, and for most power transmission

gearing, the value of this quantity lies between the values of one and two. In some instances, the contact ratio can be as high as four.

The radial length of the teeth beyond the pitch circle is called the addendum distance, and the radial depth of the teeth below the pitch circle is called the dedendum distance. By trade association standards, these distances are specified as constant multiples of the circular pitch.

The location in the gear mesh of the contact point between mating teeth can be specified conveniently by the distance, s, between the pitch point and the contact point, measured along the line of action.

When load is being transmitted through the gear mesh, the load is carried either by one pair of teeth alone or jointly by two or more pairs of teeth depending on the value of the contact ratio. It is assumed here that the contact ratio is between one and two. As the gears rotate, the load is transferred from teeth that are in mesh to succeeding teeth that are moving into mesh. Similarly, teeth moving out of mesh relinquish load as they leave contact.

2.3. Load-Deflection Properties of a Gear Mesh

2.3.1 Definition of Spring-Stiffness of Gear Mesh

The ideal curves that are used to form gear teeth are designed to produce a constant speed ratio. That is , the gears behave like two imaginary pitch cylinders which roll without slipping.

In actual gears, the materials employed cannot be absolutely rigid; consequently, the gear-teeth will deflect due to the transmitted loads,which will cause the ideal pitch circles to slip. Thus a deviation from ideal kinematic operation occurs.

For example, in Figure 2, gear 2 is fixed, by definition its pitch circle is also fixed. Now consider a torque τ_1 to be applied to the mating gear 1. This torque on gear 1 must be balanced, for static operation, by the moment of the resultant force F, which, in the absence of friction, acts along the pressure line.

$$\tau_1 = F \cos\theta R_1 \tag{2.7}$$

or in terms of T, the component of W which acts tangentially to the pitch circle,

$$\tau_1 = T R_1 \tag{2.8}$$

When friction is present, or contact between mating teeth lies off the pressure line, W and T in Equations. (2.7) and (2.8) no longer represent tooth loads exactly, but are still convenient ways of expressing the input torque τ_1.

The spring stiffness of the gear mesh is defined as the amount of tangential load T, computed from Equation (2.8), to produce one unit of pitch-circle slip, δ, as shown in Figure 2

$$K_p = \frac{T}{\delta} \tag{2.9}$$

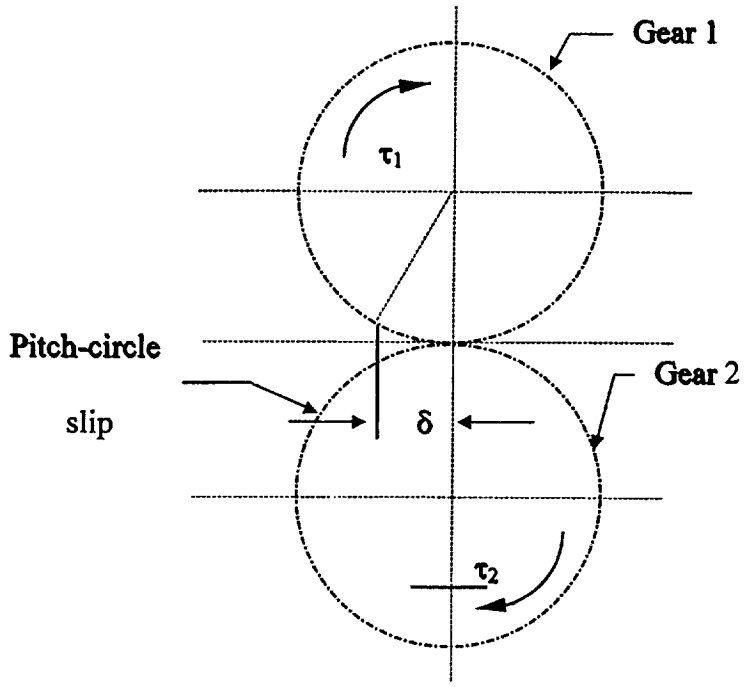
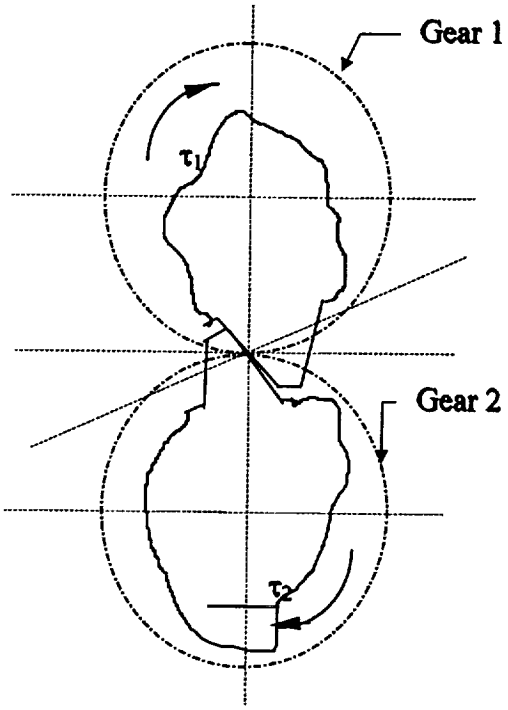
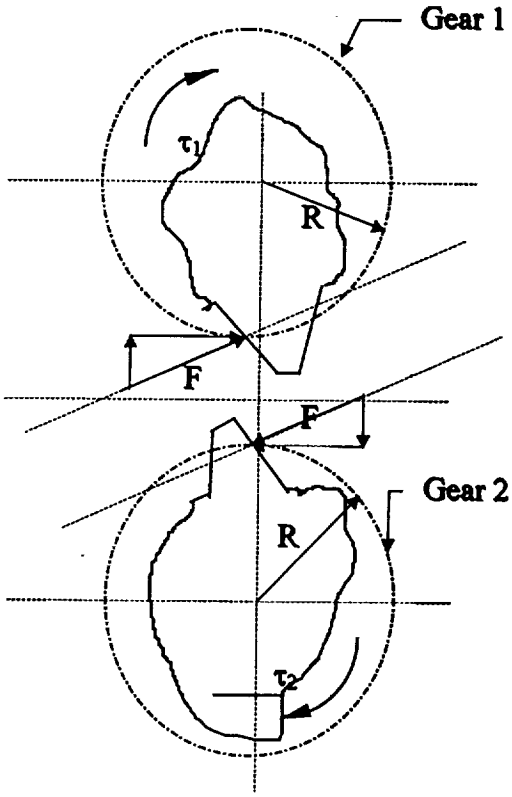


Figure 2 (a) Pitch - Circle Slip



(b) Loaded Gear Teeth



(c) Applied Loads

Figure 2 Definition of Gear-Mesh Spring-Stiffness

This definition can be equally well stated as the amount of load W acting along the pressure line, required to produce one unit of relative displacement(s_r) between gears, measured along the pressure line

$$k = \frac{F}{s_r} \tag{2.10}$$

Where

$$F \cos\theta = T \tag{2.11}$$

and

$$s_r = \delta \cos\theta \tag{2.12}$$

These two spring stiffness are related by virtue of Equations. (2.11) and (2.12)

$$k = \frac{T}{\delta(\cos^2\theta)} = \frac{K_p}{\cos^2\theta} \tag{2.13}$$

2.3.2 Compliance for a Single Pair of Teeth

The determination of the compliance of gear teeth is considerably difficult because it is an integral function of the entire loaded tooth. In addition, because of the stubbiness of the teeth, the foundation and shear effects are important. Cornell's method [20] parallels, to a great extent, Weber's work[21], however Cornell uses O'Donnell's foundation factors [22,23]. The total compliance or flexibility of a gear tooth at the point of load, y_T, is made up of three deflections:

1. The basic deflection of the tooth as a cantilever beam, y_B;
2. The deflection of the tooth caused by the fillet and foundation flexibility [22], y_F;
3. The local deflection caused by the contact and compression between the two teeth, y_L.

When two gear teeth are in contact, the total compliance is their combined deflection per unit of load at the contact position or

$$C=(y_{T1}+y_{T2})/L=[(y_{B1}+y_{B2})+(y_{F1}+y_{F2})+y_L)]/L \qquad (2.14)$$

where L is the applied load, and the deflections y are in the direction of the load. The methods to calculate the three types of deflections in equation (2.14) are given below.

2.3.2.1 The Basic Deflection of the Tooth as a Cantilever Beam

When a load L is applied to the surface of a gear tooth, a deflection of the tooth occurs in the direction of the load. Suppose that the tooth is rigid near the point of loading. Then deflections of the tooth will still occur due to each of the following effects.

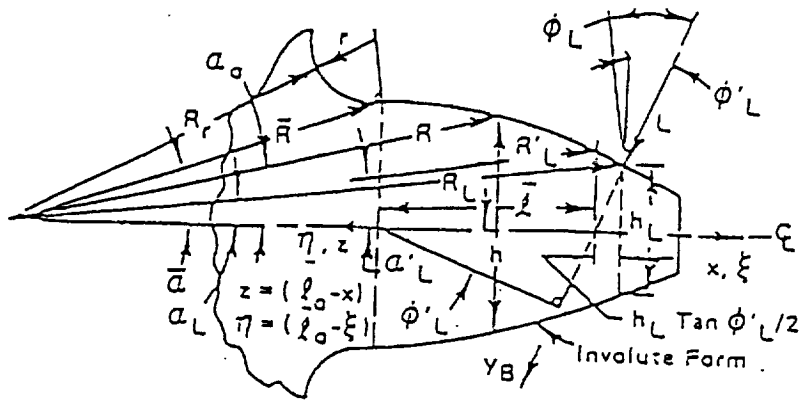
1. Bending of the tooth in the manner of a cantilever beam.
2. Direct compression of the tooth due to the radial component of the load (N)
3. Direct shearing of the tooth due to the tangential component of the load (T).

The deflection and , therefore, compliance of a gear tooth over its beam portion is easily obtained using elementary strength of materials. Referring to Figure 3(a), the total of the bending and shear deflection in the direction of and at the point of application of the tooth load L, which is at radius R_L or position S along the line of action, can be expressed in integral form as

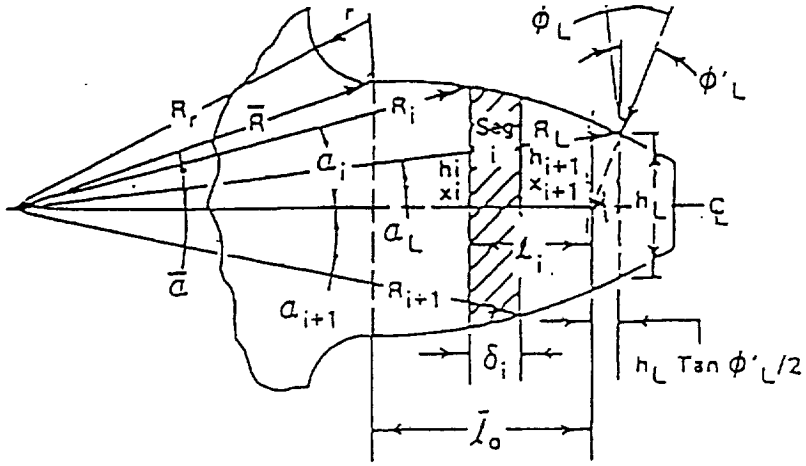
$$y_B=\frac{L\cos^2\phi_L'}{E}\int_0^{l_0}\{\int_z^{l_0}\frac{\eta}{I}d\eta+\frac{2.40(1+\mu)+Tan^2\phi_L'}{A}\}dz \qquad (2.15)$$

where I is the section modulus of the tooth, and using the relationship of

$$\xi=x \quad \text{or} \quad \eta=z; \quad \eta=(\bar{l}-\xi)=z=(\bar{l}-x) \qquad (2.16)$$



(a) Integration Form



(b) Segmental Form

Figure 3. Beam compliance of a gear tooth

and a value of 1.2 has been assumed for the shear factor based on a rectangular tooth;

$$G = E/2(1+\mu) \tag{2.17}$$

and A is the cross-sectional area as a function of x or z. The deflection of the basic tooth can also be defined in a summation expression rather than integral form, see Figure 3(b), In this case the tooth beam deflection at and in the direction of the load is

$$y_B = \frac{L\cos^2\phi_L'}{E}\sum_{i=1}^{n}\delta_i\left\{\frac{\left(l_i^2 - l_i\delta_i + \frac{1}{3}\delta_i^2\right)}{\bar{I}_i} + \frac{\left(2.4(1+\mu) + Tan^2\phi_L'\right)}{\bar{A}_i}\right\} \tag{2.18}$$

Where

$$1/\bar{I}_i = \left(1/I_i + 1/I_{i+1}\right)/2 \tag{2.19}$$

and

$$1/\bar{A}_i = (1/A_i + 1/A_{i+1})/2 \tag{2.20}$$

Using these inverse forms for the values of $\bar{I}_i$ *and* $\bar{A}_i$ improves the accuracy for a small number of elements. In equation (2.18)

$$l_i = (\bar{l} - x_i) \tag{2.21}$$

and

$$\delta_i = (x_{i+1} - x_i) \tag{2.22}$$

Both approaches for beam flexibility assume a narrow tooth width, W. For wide teeth where $W/h_p > 5$, the flexibility is decreased by the antielastic effect, so that the values of I in equations (2.15) and (2.18) should be divided by $(1-\mu^2)$.

2.3.2.2 The Deflection of the Tooth Caused by the Fillet and Foundation Flexibility

Because of the fillet and the flexibility of the material to which the tooth is attached, additional deflection will occur at the load; see references [21- 23]. This fillet and foundation deflection in the direction of the load, y_F, is a function of the fillet geometry and the load position and direction, and is determined by the effective fillet length or angle γ_F for which the maximum deflection or work occurs at the load.

Based on Figure 4, O'Donnell [22, 23] shows that the deflection at and in the direction of the tooth load due to the foundation effects, y_{FF}, for plane stress is

$$y_{FF} = \frac{LCos^2\phi_L'}{WE}\left[\frac{16.67}{\pi}\left(\frac{l_F}{h_F}\right)^2 + 2(1-\mu)\left(\frac{l_F}{h_F}\right) + 1.534\left(1+\frac{Tan^2\phi_L'}{2.4(1+\mu)}\right)\right] \qquad (2.23)$$

For wide teeth the expression for plane strain is used or

$$y_{FF} = \frac{LCos^2\phi_L'}{WE}(1-\mu^2)\left[\frac{16.67}{\pi}\left(\frac{l_F}{h_F}\right)^2 + 2\left(\frac{1-\mu-2\mu^2}{1-\mu^2}\right)\left(\frac{l_F}{h_F}\right) + 1.534\left(1+\frac{Tan^2\phi_L'}{2.4(1+\mu)}\right)\right] \qquad (2.24)$$

The O'Donnell coefficients in equations (2.23) and (2.24) differ slightly from those given by Weber[21]. The first term in the brackets is the deflection at L due to the rotation caused by the moment at h_F. The second term is the sum of the deflections at L due to the displacement at h_F caused by the moment at h_F and the rotation at h_F caused by the shear force at h_F. The first part of the third term is the displacement at L due to the shear force at h_F based on the assumption that the effective depth for determining this deflection is $2^1/_2$ times the tooth thickness [21]. The second part of the third term is the deflection at

L due to the normal component of the load assuming the same relationship holds as indicated by the beam deflection equation; see equations (2.15) and (2.18).

Referring to Figure 4, the deflection at and in the direction of the load due to the flexibility of the fillet and foundation, y_F, the shaded region, is obtained from the summation of the fillet beam deflection, y_{FB}, using equations (2.15) and (2.18), and the foundation deflection y_{FF}, using equation (2.23) or (2.24), i.e.,

$$y_F = y_{FF} + y_{FB} \tag{2.25}$$

where

$$\begin{aligned} l_F &= \bar{l} + r(Sin\gamma_F - Sin\bar{\gamma}) \\ h_F &= \bar{h} + 2r(Cos\bar{\gamma} - Cos\gamma_F) \\ l_i &= \bar{l} + r(Sin\gamma_i - Sin\bar{\gamma}) \\ h_i &= \bar{h} + 2r(Cos\bar{\gamma} - Cos\gamma_i) \end{aligned} \tag{2.26}$$

The value of γ_F is the one that maximizes the value of y_F or y_T, which can easily be done as one integrates or sums up the deflection of the tooth starting at the beginning of the fillet or at the load position.

2.3.2.3 The Local Deflection Caused by Compression of the Tooth Surfaces

The local compliance, y_L, consists of the Hertz or line contact deflection plus the compression of each tooth between the point of contact and the tooth centerline. Figure 15 gives the nomenclature for the parameters that determine the local deformation. There are several approaches to calculate the local compliance. All of the expressions for the local deformation are nonlinear with load because of the Hertz half contact width b. Here, the closed form approach developed by Weber[21] was adopted.

Weber in reference[21] developed an expression specifically for the local deformation of two gear teeth, based on Hertz's work on deformation between cylinders. In order to obtain a closed form solution, he assumed small deformations so that just the first two terms of the binomial expansion of the deformation needed to be used - i.e.,

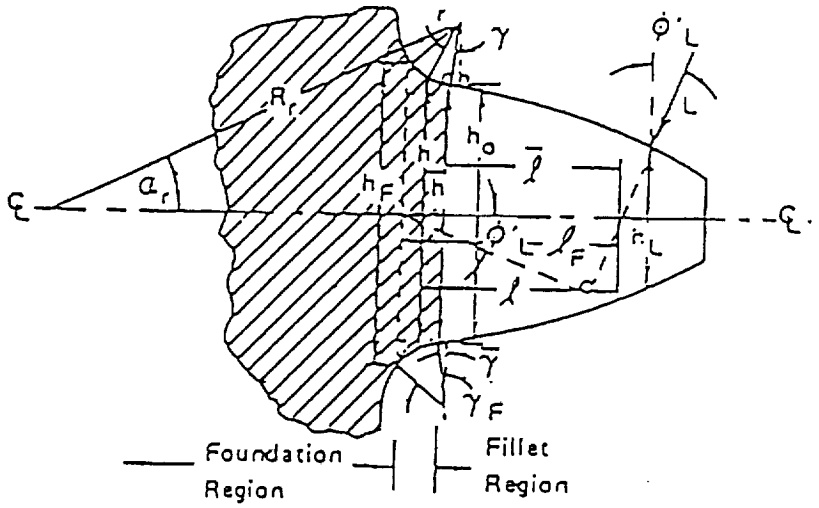


Figure 4. Fillet and foundation compliance of a gear tooth

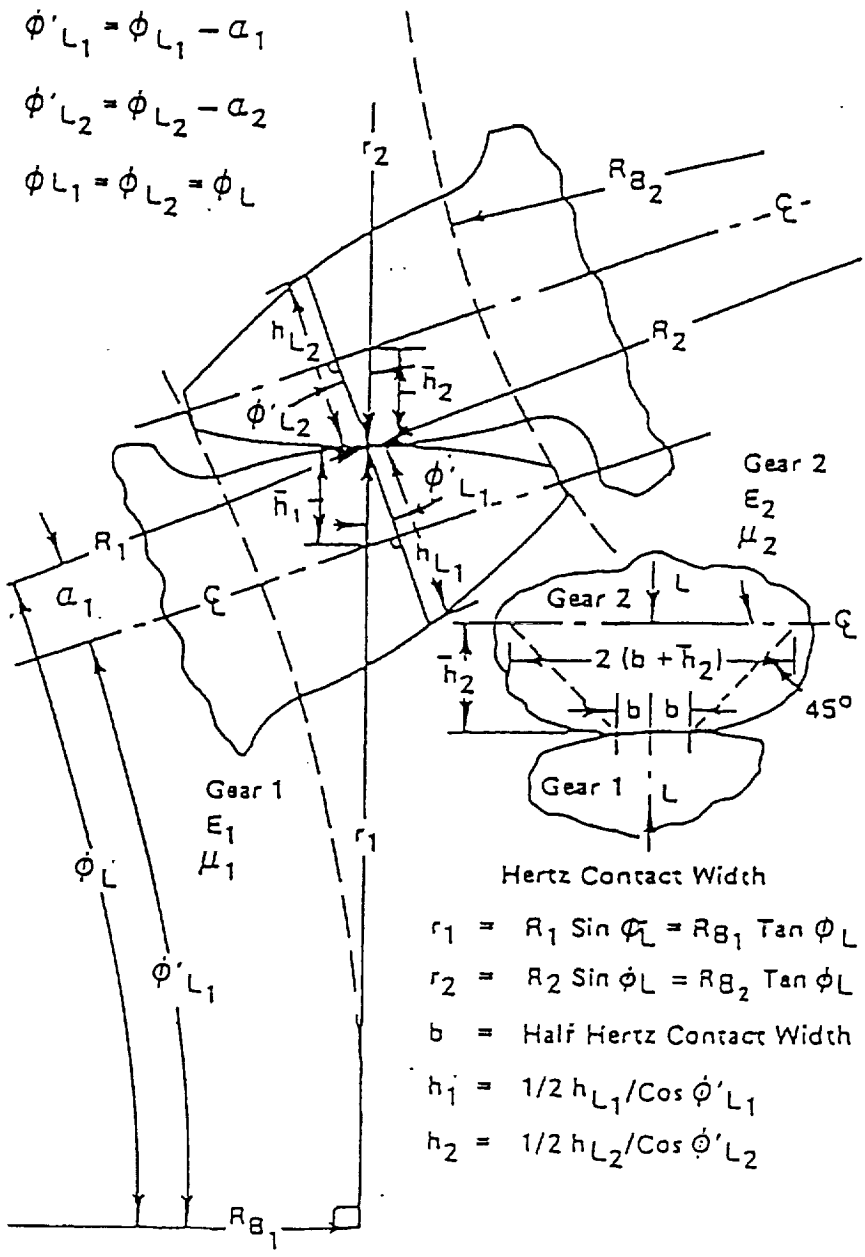


Figure 5. Nomenclature for local compliance

$$y_L = \frac{2L}{\pi W}\left[\left(\frac{1-\mu_1^2}{E_1}\right)\left\{\ln\frac{2\bar{h}_1}{b} - \left(\frac{\mu_1}{2(1-\mu_1)}\right)\right\} + \left(\frac{1-\mu_2^2}{E_2}\right)\left\{\ln\frac{2\bar{h}_2}{b} - \left(\frac{\mu_2}{2(1-\mu_2)}\right)\right\}\right] \qquad (2.27)$$

where

$$b = \left\{ \frac{4L}{\pi W} \left[\left(\frac{1-\mu_1^2}{E_1} \right) + \left(\frac{1-\mu_2^2}{E_2} \right) \right] \Big/ \left[1/r_1 + 1/r_2 \right] \right\}^{1/2} \qquad (2.28)$$

If $E_1 = E_2$ and $\mu_1 = \mu_2$, equation (2.27) reduces to

$$y_L = \frac{4(1-\mu^2)}{\pi E} \frac{L}{W} \left[\ln\left(2\frac{\sqrt{\bar{h}_1 \bar{h}_2}}{b} \right) - \left(\frac{\mu}{2(1-\mu)} \right) \right] \qquad (2.29)$$

The overall compliance, C, of the tooth pair is obtained by adding the local gear tooth compliance defined by equation (2.27) with the gross tooth compliance for the two gear teeth

$(y_B + y_F) / L$ given by equations (2.15) or (2.18) and (2.25); see equation (2.14).

2.3.3 Load - Deflection Relationship of a Gear Mesh

When the total compliance for a single pair of teeth is known for all points of contact along the pressure line, computation of the gear-mesh compliance can be carried out. Two cases are possible:

1. Only one pair of teeth is in contact; then the mesh compliance is equal to the single - tooth pair compliance. This is normally the case when contact is near the pitch point.
2. More than one pair of teeth is in contact. Successive pairs of teeth are spaced along the line of action by one normal pitch. Consequently, the compliance curves for successive tooth -pairs also can be spaced along the line of action by the same amount.

For simplicity, the real gear - action is modeled as shown in Figure 6 which represents engagement and disengagement of the various pairs of teeth. The gear action can be represented by the movement of a cam underneath successive pairs of teeth. This

cam is shown in Figure 6 during its passage underneath tooth-pairs n, n-1 and n+1; however, after leaving these pairs, it will continue on, bringing other tooth pairs into contact in succession. Point P on the cam surface corresponds to pitch-point contact, and when any tooth-spring is contacting the cam at P, that tooth-pair is in contact at the pitch-point in the real gear system. Tooth-springs in the model are spaced in the gear by an interval of one normal pitch, just as the actual tooth-pairs are spaced in the gear system. The distance AC on the cam is flat, and is equal to the ideal path of contact. Any tooth-pair start engagement at point E and is in full engagement at point A and remain in contact until point C is reached, then starts disengagement, untilpoint D is reached.

The load W is the total load transmitted through the mesh, and s_r is the relative motion between mating gears, measured along the pressure line. The relative motion is resisted by the tooth-pair stiffness, k_n , k_{n+1} , k_{n-1} , etc., depending on the number of teeth in contact.

For example, tooth-pairs n and n+1 are in contact at the position shown in Figure 6 , the compliance for the mesh at this position is given by the resultant compliance of the

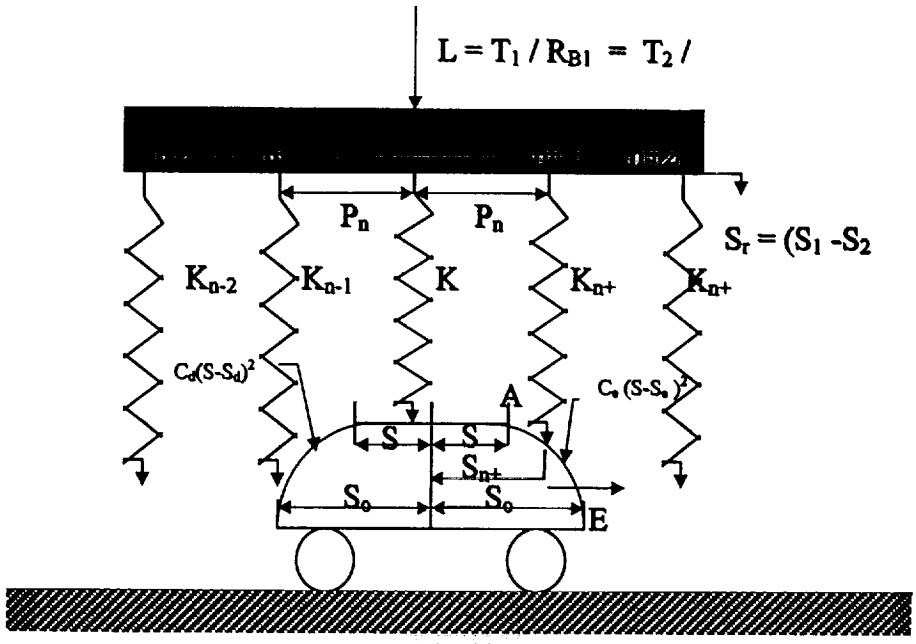


Fig. 6 Model of Gear Action

tooth-springs n and n+1. Since the load L is shared between the two springs, the total compliance is the parallel-combination of the individual compliance.

$$\frac{1}{C_{total}} = \frac{1}{C_n} + \frac{1}{C_{n+1}} \tag{2.30}$$

or

$$C_{total} = \frac{C_n C_{n+1}}{C_n + C_{n+1}} \tag{2.31}$$

This result is easily generalized to the combined compliance for n pairs of teeth.

$$C_n = \frac{\prod_1^n C_i}{\sum_{r=1}^{n} \left[\prod_{k=1}^{r-1} C_k \prod_{m=r+1}^{n} C_m \right]} \tag{2.32}$$

where each C refers to the component compliance of a given tooth-pair at a certain location along the pressure line.

2.4. Model of Imperfection in Gear Tooth

To simulate gear failures, a computer code developed at NASA Lewis Research Center[10] was modified to simulate various types of gear tooth imperfection. The imperfection of the gear tooth was modeled by using a tooth profile modification procedure that simulates changes in gear tooth surface profile. The tooth profile modification is represented by their respective cams in Figure 6. The cams representing the tooth profile modifications have the form

$C_e (S - S_e)^x$ and $C_d (-S_d - S)^x$ respectively (2.33)

where x is an integer. Typical gear tooth profile modifications were found to be represented quite well by simply using a cam form with x = 2; see Figure 6.

Hence the imperfection of the tooth can be simulated by engagement and disengagement tooth profile modifications ε_d and ε_e, which represent the deviation of the profile from the ideal tooth geometry, and are assumed to be a square function of the distance along the line of action, e.g.

$$\varepsilon_{je} = C_e(S_j - S_e)^2 \quad \text{and} \quad \varepsilon_{jd} = C_d(S_j - S_d)^2 \tag{2.34}$$

where S_j reflects the distance along the line of action from the pitch line; S_d and S_e, the distance along line of action from pitch line to the start of disengagement cam and end of engagement cam; C_d and C_e, disengagement and engagement cam relief, which are determined by:

$$C_e = \Delta_e / (S_{oe} - S_e)^2 \quad \text{and} \quad C_d = \Delta_d / (S_{od} - S_d)^2 \tag{2.35}$$

Where S_{od} and S_{oe}, represent distances along the line of action from pitch line to end of disengagement cam (tooth tip) and start of engagement cam; Δ_d and Δ_e, maximum disengagement and engagement tooth relief for the tooth pair, which is the total or combined tooth pair relief at the start and end of an ideal mesh.

Figure 7 shows how this model in Figure 6 relates to the real gear tooth. Suppose the tooth shown in Figure 7 is the driven gear. In Figure 7, the point P is the pitch point of the tooth pair; point E is the start point of engagement; point A is the end point of engagement; point C is the start point of disengagement; and point D is the end point of disengagement.

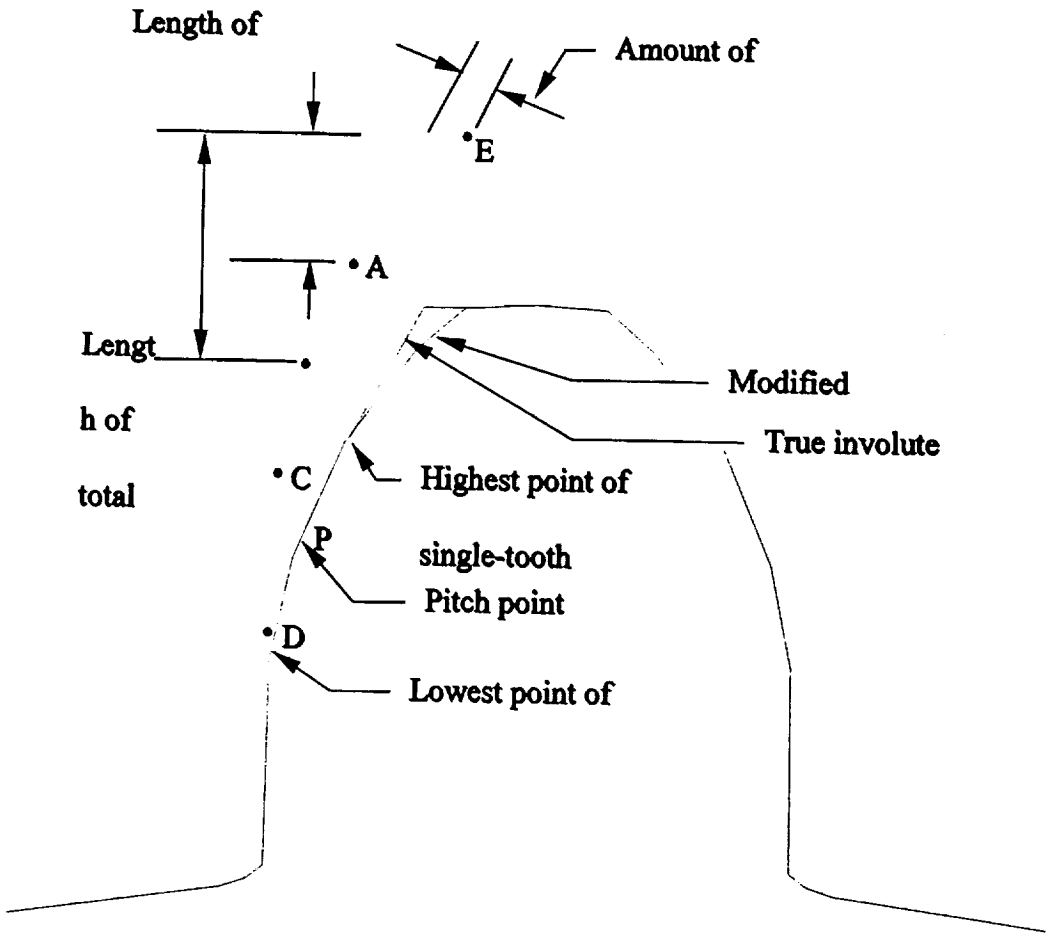


Figure 7. Gear tooth with modified tooth profile

(The lengths of S_{oe} and S_e are measured along the line of action)

Tooth profile modification can be divided into two parts: one is the engagement part, which is the tooth face between points E and A; another is the disengagement part, which is the tooth face between points C and D. When two gears are in mesh, modifying the tooth tip of one gear is the same as modifying the tooth root of the other gear.

Hence the disengagement part can be defined as the driving gear tooth face between point A and E.

To simulate the wear of a tooth face, two parameters needed to be specified. One is P_{SOD} / P_{SOE} , which is the length of profile modification as a percent of the length of total disengagement or engagement measured along the line of action. This parameter determines the length of tooth wear. Another one is Δ_d and Δ_e , which is the maximum disengagement and engagement tooth relief for the tooth pair, which is the total or combined tooth pair relief at the start and end of an ideal mesh. This parameter determines the maximum wear at the tip or the root of the gear tooth.

2.5. Discussions of Results

As outlined above, there are two parameters which control the simulated tooth wear. A parametric study was conducted to evaluate the change in gear-mesh stiffness under different wear conditions. The analysis given above, along with its corresponding computer code was used to analyze the gear-mesh stiffness change of a gear system under different wear conditions.

For simplicity, a single stage gear mesh was chosen (one input gear and one output gear). The pertinent parameters for the system are given in Table 1. An example of the input data for the system is given in Table 2.

Table 1 Parameters of Gear System

	input gear	output gear
Diametrical Pitch	8.4667	8.4667
Pressure Angle (degree)	22.5	22.5
Number of Teeth	14	28
Face Width (in)	1.1811	1.1811
Input Torque (lb.in)	30	
Input RPM	7000	

Table 2 Gearbox Input Data Set

```
Gearbox, Input pinion & gear
5     1.     2.     8.4667     22.5     0.
0.
2     6.     14.    28.
2     9.     1.1811 1.1811
3     12.    1.     4.     30.
3     15.    6000.  6000.    1.
1     28.    .0280835
1     31.    .077535
1     52.    .1
4     60.    50.    50.    .000021  .000021
3     66.    60.    40.    .00001
1     78.    .00001
1     120.   0.
3     140.   25.    180.   .0528
2     150.   .01    20.
1     167.   .0075
3     481.   .00000000 .00000000 .00000000
3     521.   .00000  .0000  .00000
1     699.   0.
1     806.   1.
0-1.
```

First, Let's analyze the wear of engagement:

1. P_{soe} was fixed at 50%, that is, the length of wear was fixed; the tip relief, Δ_e, was chosen as 1e-5 in, 2e-5 in, and 4e-5 in. The results are shown in Figure 8.
2. Δ_e was fixed at 2e-5 in, that is, the maximum amount of wear at the tip or the root of the gear tooth was fixed; the length of wear, P_{soe}, varied at 50%, 60%, 70% and 80%. The results are shown in Figure 9.

Second, Let's analyze the wear of disengagement part:

1. P_{sod} was fixed at 50%, that is, the length of wear was fixed; the tip relief, Δ_d, was chosen as 1e-5 in, 2e-5 in, and 4e-5 in. The results are shown in Figure 10.
2. Δ_d was fixed as 2e-5 in, that is the maximum amount of wear at the tip or the root of the gear tooth was fixed; the length of wear, P_{sod}, varied as 50%, 60%, 70%, and 80%. The results were shown in Figure 11.

Finally, both the wear of the engagement and disengagement were considered:

1. P_{soe} and P_{sod} were fixed at 50%, that is the length of wear was fixed; the tip relief, Δ_e and Δ_d, were chosen as 2e-5 in, 4e-5 in, 6e-5 in, and 8e-5 in. The results are shown in Figure 12.
2. Δ_e and Δ_d were fixed at 2e-5 in, that is the maximum amount of wear at the tip or the root of the gear tooth was fixed; the length of wear, P_{soe} and P_{sod}, varied as 50%, 60%, 70%, and 80%. The results are shown in Figure 13.

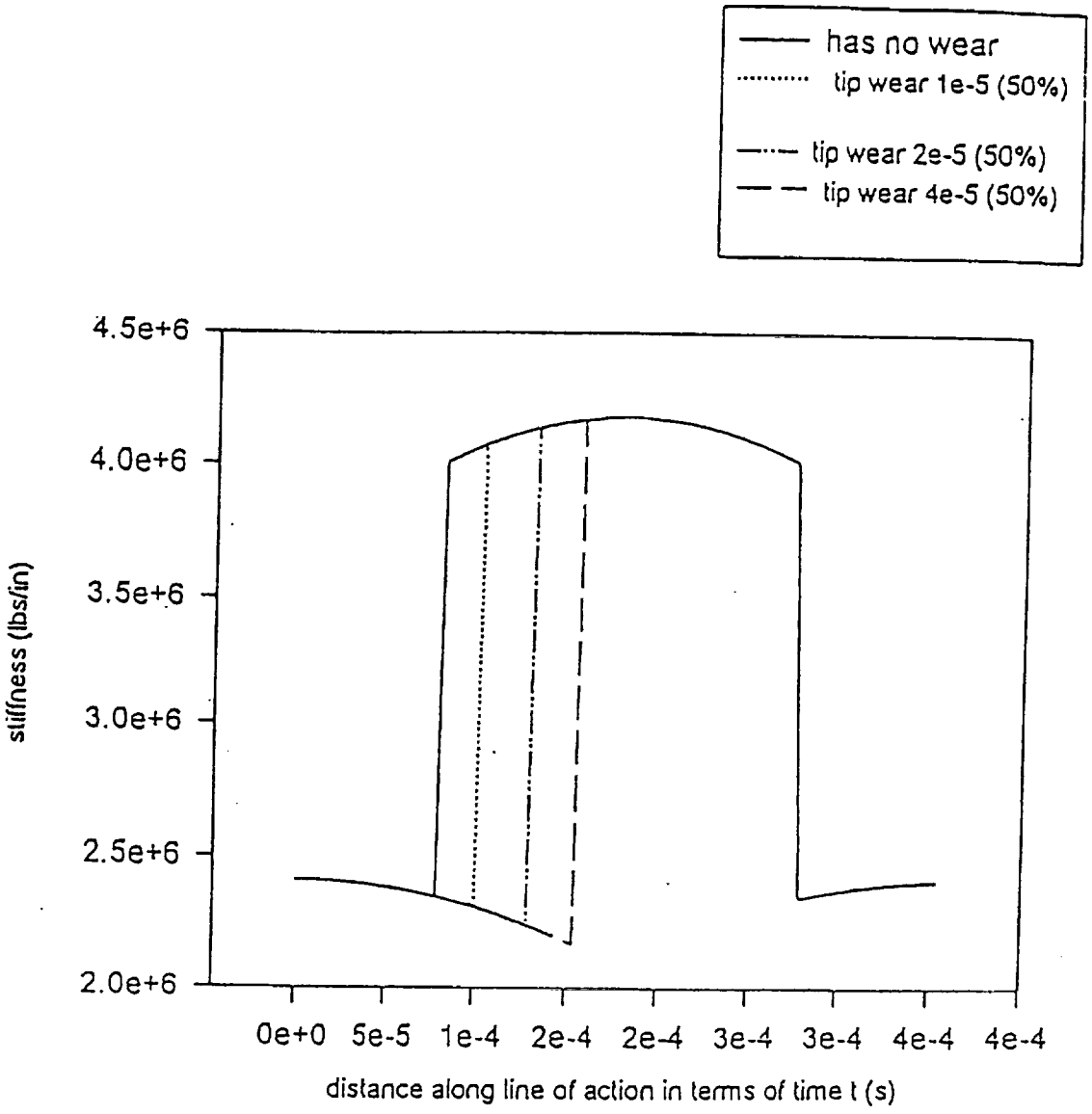


Figure 8. Mesh stiffness due to the tooth surface wear on engagement

(Different value of tip relief, while the wear length remains fixed)

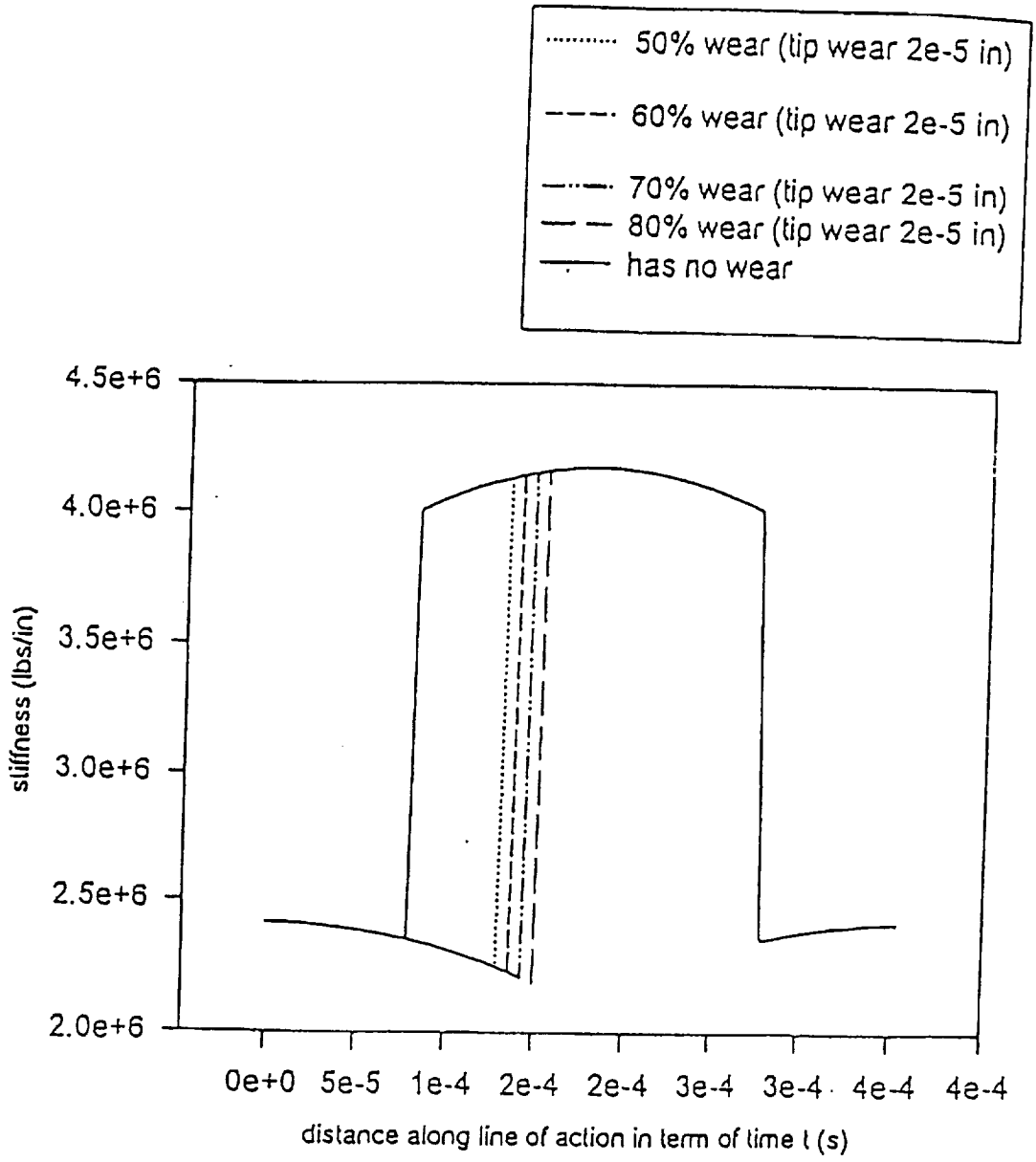


Figure 9. Mesh stiffness due to tooth surface wear on engagement

(Different value of wear length, while the tip relief remains fixed)

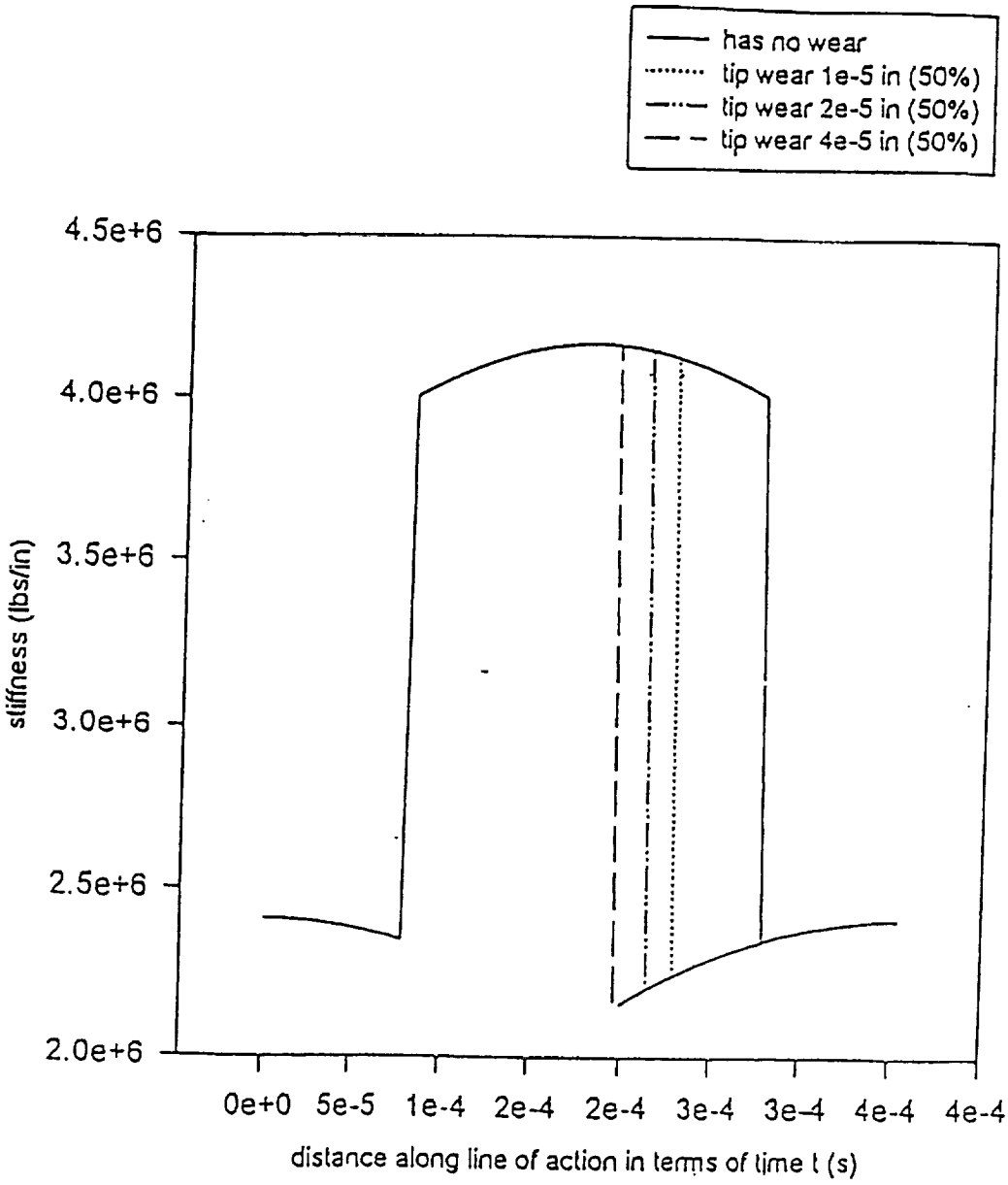


Figure 10. Mesh stiffness due to tooth surface wear on disengagement

(Different value of tip relief, while the wear length remains fixed)

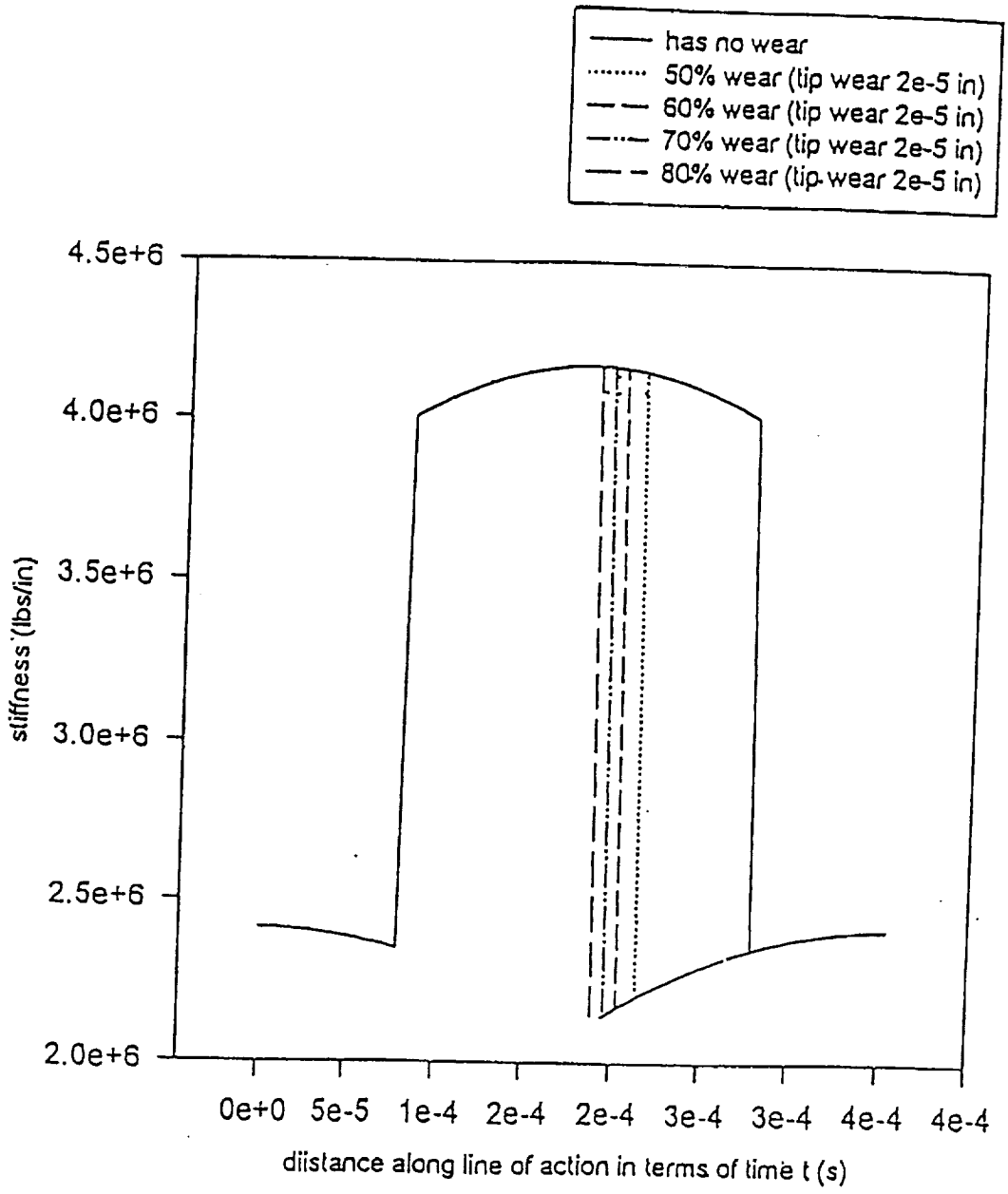


Figure 11. Mesh stiffness due to tooth surface wear on disengagement (Different value of wear length, while the tip relief remains fixed)

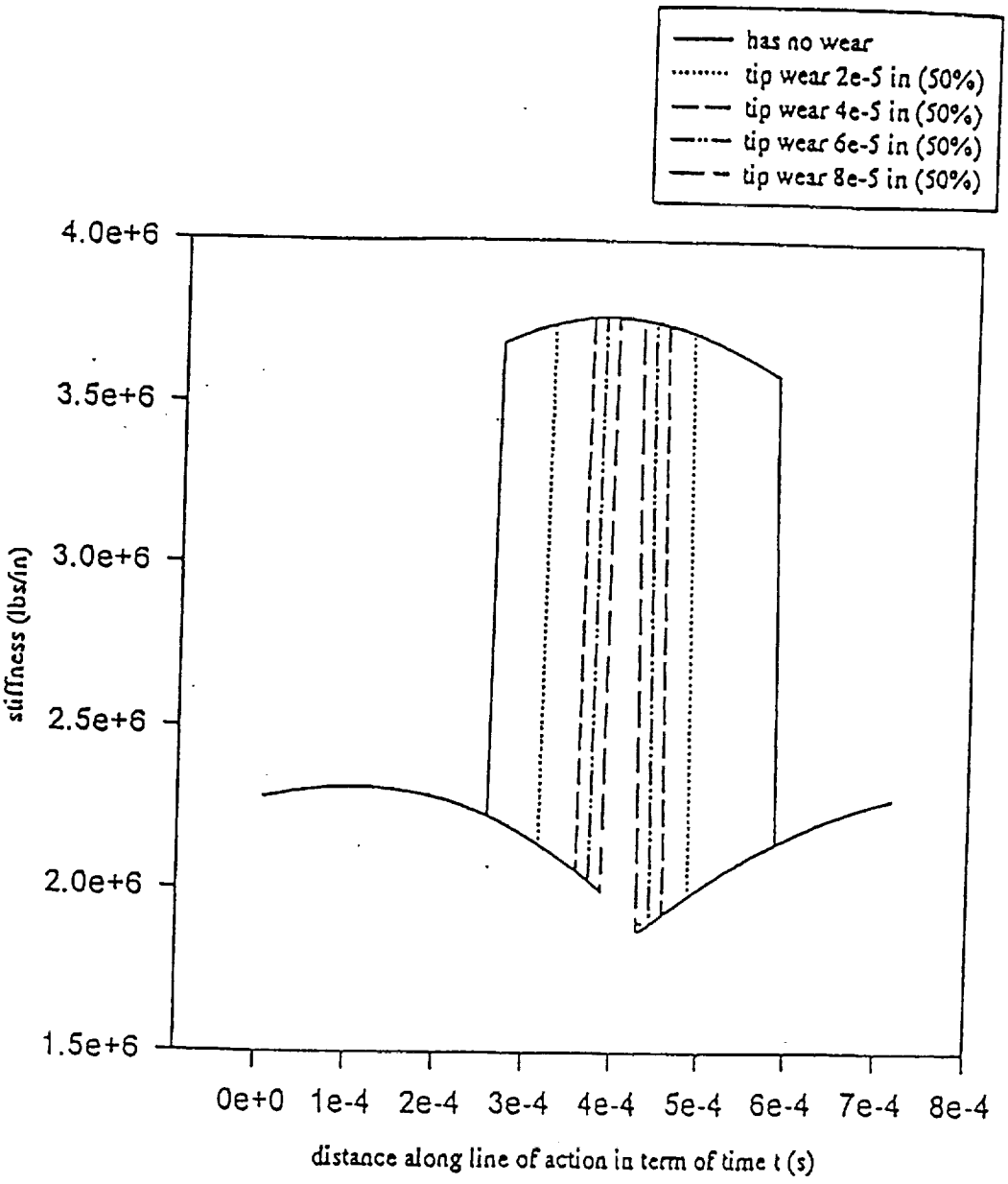


Figure 12. Mesh stiffness due to tooth surface wear on both engagement and disengagement (Different value of tip relief, while the wear length remains fixed)

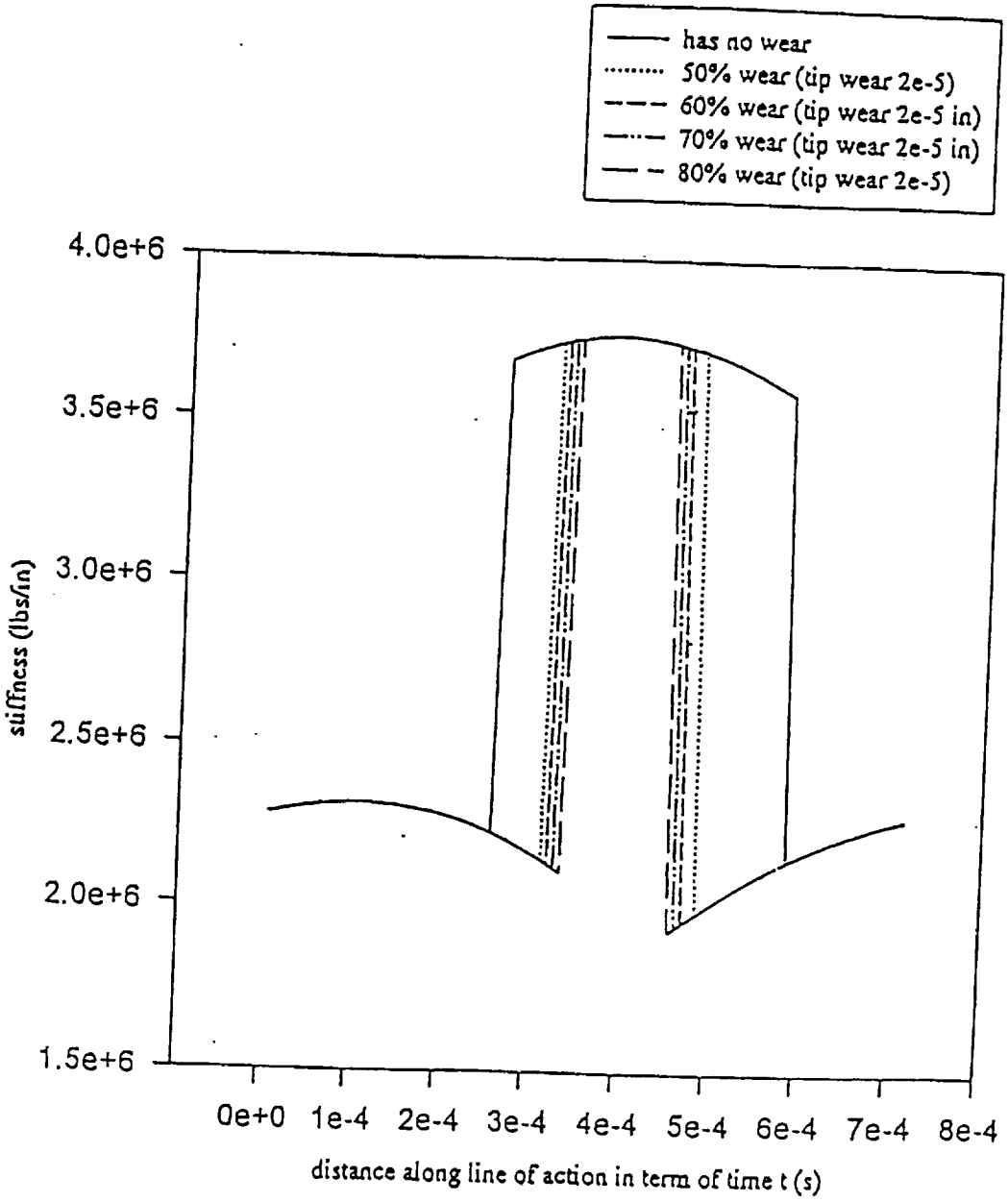


Figure 13. Mesh stiffness due to tooth surface wear on both engagement and disengagement (Different value of wear length, while tip relief remains fixed)

From Figure 8-13, it is obvious that the gear tooth damage due to surface pitting and wear can significantly change the phase of the mesh stiffness. The higher degree of surface pitting and wear , the more the phase of the mesh stiffness will shift.

2.6. Summary

A numerical model has been developed to simulate the gear mesh stiffness change resulting from gear tooth damage due to surface pitting and wear. The work in this chapter can be summarized as follows:

1. A method has been developed to simulate the tooth surface wear in a gear transmission system. The tooth surface wear level can be controlled by adjustment of both amplitude and length in the tooth profile modification.
2. The gear mesh model has been developed to provide the mesh stiffness with the effect of gear tooth damage due to various degrees of surface pitting and wear. It will be shown in the next chapter that these changes in gear mesh stiffness can be incorporated into a dynamic simulation of the gear transmission system for dynamic predictions.

CHAPTER 3

THE EFFECTS OF GEAR TOOTH IMPERFECTION ON THE DYNAMIC CHARACTERISTICS OF GEAR TRANSMISSION SYSTEMS

3.1. Objectives

In the last two decades, problems arising from excessive gear tooth wear and gear tooth surface pitting in gear transmission systems have been of increasing concern for a variety of gear users. Although regular visual inspections and preventive maintenance can help to reduce the failure rate of gear systems, the cost and down time required make such programs inefficient and uneconomical.

Vibration signature analysis methodologies are being developed to non-intrusively examine the health and wear of gear transmission systems. Considerable success has been achieved in applying the Wigner-Ville distribution concept (WVD) [14-16] in machine health monitoring of gear transmission systems [17,18,24]. However , a complete vibration signature database is necessary for the development of an effective pattern recognition scheme. In order to populate such databases, the development of an accurate analytical procedure to predict vibrations in gear systems due to wear and fatigue failure is necessary.

The objective of this chapter is to develop a comprehensive procedure to simulate and analyze the vibration in a gear transmission system with effects of surface pitting and wear of the gear teeth under normal operating conditions. To simulate the vibration of the transmission system, the equations of motion were established individually for each rotor-gear-bearing system. The changes of the mesh stiffness at one particular tooth

location or a number of consecutive teeth due to the effects of surface pitting and wear were incorporated into the gear-rotor model for the dynamic simulation [11-13]. The dynamics of each gear-rotor system were coupled with each other through the gear mesh interacting forces. The coupling between the rotors and the casing structure were joined through the bearing support forces. The global vibrations of the system were evaluated by solving the transient dynamics of each rotor system simultaneously with the vibration of the casing. In order to minimize the computational effort, the number of degrees-of-freedom of the system were reduced by using a modal synthesis procedure [11,12]. The results were evaluated by Wigner-Ville Distribution (WVD) [13-15].

3.2. Dynamics of the Gear-Shaft Configuration and the Gearbox System

The dynamics of the ith individual gear-shaft system can be evaluated through the equations of motion for the vibrations of an individual rotor-bearing-gear system as shown in Figure 14, given in matrix form, as

$$[M]\{\ddot{W}_i\}+[K_s]\{W_i\}=\{F_{bi}(t)\}+\{F_{gi}(t)\}+\{F_{ui}(t)\} \tag{3.1}$$

where [M] and $[K_s]$ are respectively the mass and shaft stiffness matrices of the rotor, $\{W_i\}$ is the general displacement vector of the ith rotor in the its local coordinate system, and, $\{F_{bi}(t)\}$, $\{F_{gi}(t)\}$, and $\{F_{ui}(t)\}$ are the force vectors acting on the ith rotor system due to bearing forces, gear mesh interactions, and mass-imbalances respectively.

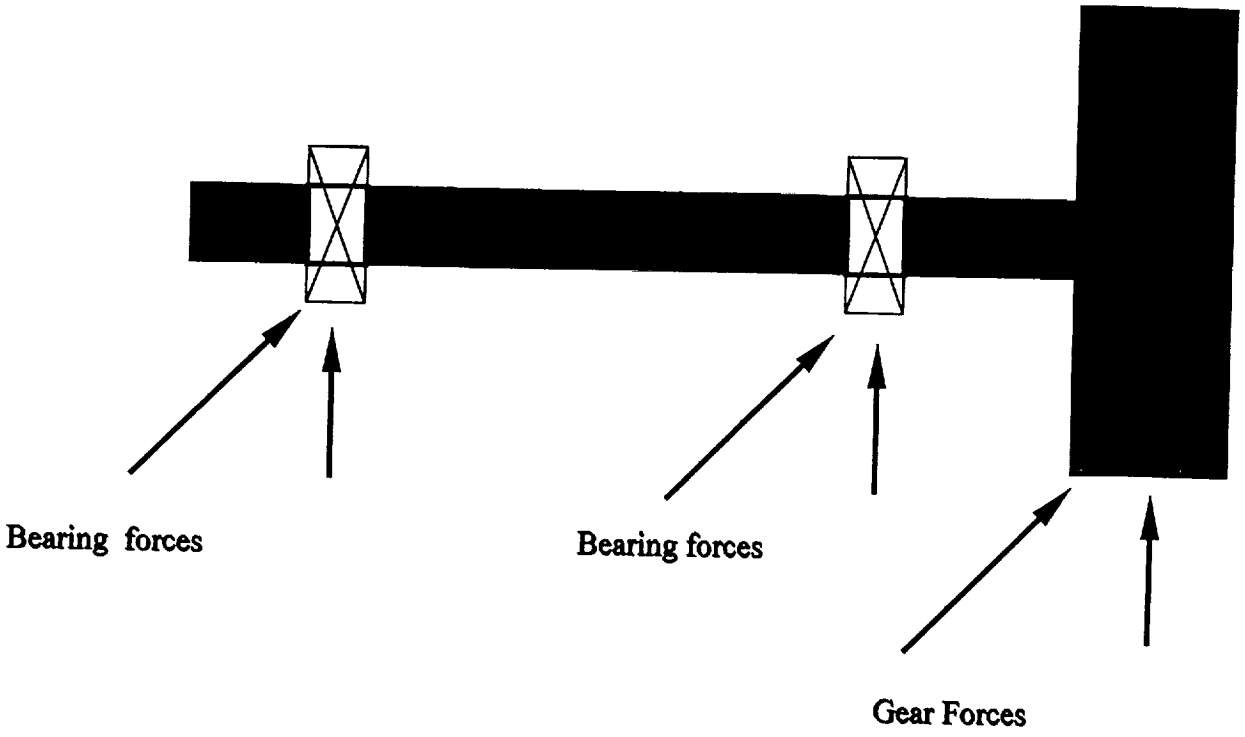


Figure 14. Schematic of a rotor-gear bearing system

The equation of motion of the gearbox with p rotor systems can be expressed as
Figure 14. Schematic of a rotor-gear bearing system.

$$[M_c]\{\ddot{W}_c\}+[K_c]\{W_c\}=\sum_{i=1}^{p}[T_{ci}]\{F_{bi}(t)\} \qquad (3.2)$$

where $[T_{ci}]$ represents the coordinate transformation between the ith rotor and the gearbox.

The bearing forces $\{F_{bi}(t)\}$ for the ith rotor can be evaluated as

$$\{F_{bi}(t)\} = [C_{bi}](\{\dot{W}_i\} - [T_{ic}]\{\dot{W}_{ci}\}) + [K_{bi}](\{W_i\} - [T_{ic}]\{W_{ci}\}) \quad (3.3)$$

where $[C_{bi}]$ and $[K_{bi}]$ are respectively the damping and stiffness of the bearing, $[T_{ic}]$ is the coordinate transformation matrix for the gearbox with respect to the ith rotor, and W_{ci} are the casing displacements at the rotor locations.

The gear forces generated from the gear mesh interaction[40] can be written as

$$\{F_{gi}(t)\} = \{F_{ri}(t)\} + \{F_{ti}(t)\} \quad (3.4)$$

where $\{F_{ri}(t)\}$ is the vector containing the gear forces and moments resulting from the relative rotation between the two mating gears and $\{F_{ti}(t)\}$ is the vector containing gear forces and moments due to the translational motion between the two gears.

In order to calculate the transient and steady state dynamics of the system, the coupled rotor and casing equations of motion must be solved simultaneously. To minimize the computational effort, the modal transformation [11,12] procedure is applied to reduce the degrees of freedom of the global equations of motion. Using m undamped mode shapes of the ith rotor system $[\phi_{i1}, \phi_{i2}, \phi_{i3}, \ldots, \phi_{im}]$ and m_c undamped mode shapes of the gearbox $[\phi_{c1}, \phi_{c2}, \phi_{c3}, \ldots, \phi_{cmc}]$, the rotor displacement for the ith rotor can be written as

$$\{W_i\} = \sum_{j=1}^{m} A_{ij}\{\phi_{ij}\} \quad (3.5)$$

and, similarly, the gearbox displacements as

$$\{W_c\} = [\phi_c]\{A_c\} \quad (3.6)$$

where $\{A_i\}$ and $\{A_c\}$ are the modal time functions of the ith rotor and the gearbox respectively. Using the expansion in equation (3.5), the equations of motion for the ith rotor in equation (3.1) can be written as

$$[M][\phi_c]\{\ddot{A}_i\}+[K_s][\phi_i]\{A_i\}=\{F_{bi}(t)\}+\{F_{gi}(t)\}+\{F_{ui}(t)\} \qquad (3.7)$$

Premultiplying by $[\phi_i]^T$ and using the orthogonality conditions of the mode shapes [21], the ith rotor equations of motion can be written as

$$\{\ddot{A}_i\}+[\Lambda^2]\{A_i\}=\{\overline{F}_{bi}\}+\{\overline{F}_{gi}\}+\{\overline{F}_{ui} \qquad (3.8)$$

where $[\Lambda^2]$ is the diagonal matrix of the squares of the natural frequencies of the system.

For the gearbox system, a similar transformation is carried out and equation(3.2) can be written as

$$\{\ddot{A}_c\}+[\Lambda^2]\{A_c\}=\{\overline{F}_{cb}\} \qquad (3.9)$$

For the system of k rotors, equation (3.8) can be repeated k times and solved with the casing equation (3.9) simultaneously for the modal accelerations $\{A_i\}$ and $\{A_c\}$. A numerical integration scheme is used to integrate the accelerations to obtain velocities and displacements at each time step for transient calculations [11].

3.3. Vibration Signature Analysis

3.3.1 Joint Time-Frequency Analysis

To examine the vibration signal, a joint time-frequency analysis method was chosen. This approach was chosen because of the large amount of information represented in the joint time-frequency results which can not be represented separately in either the time domain or the frequency domain. The joint time-frequency analysis will

provide an instantaneous frequency spectrum of the system at every instant of the revolution of the pinion while a Fourier Transform can only provide the average vibration spectrum of the signal obtained during one complete revolution. In other words, the time-changing spectral density from the joint time-frequency spectra will provide information concerning the frequency distribution concentrated at that instant around the excited instantaneous frequency which cannot be obtained in a regular vibration frequency spectrum. The following is a description of the joint time-frequency analysis method.

To examine the vibration signal in a joint time-frequency domain, the Wigner-Ville method [14,15] was used in this research. While the Fast Fourier Transform (FFT) technique can provide the spectral contents of the time signal, it cannot distinguish time phase change during a complete cycle of operation. In other words, it assumes that the time signals are repeatable for each time data acquisition window without considering the effects of any magnitude and phase changes during the sampling period. The Wigner-Ville distribution will provide an interactive relationship between time and frequency during the period of the time data window. The comprehensive representation of the vibration signal using the WVD method is the primary reason that it was used to compare the predicted and experimental vibration results. The WVD (Wigner-Ville Distribution), in a Discrete form, can be written as:

$$W_x(nT, f) = 2T \sum_{i=-L}^{L} x(nT + iT)x^*(nT - iT) \cdot e^{-j4\pi fiT} \tag{3.10}$$

where

W(t,f) = the Wigner-Ville distribution in both the time domain t and frequency domain f.

x(t) = the time signal

T = the sampling interval

L = the length of time data used in the transform

To allow sampling at the Nyquist rate and eliminate the concentration of energy around the frequency origin due to the cross product between negative and positive frequency [14,15], the analytic signal was used in evaluating the WVD. The analytic signal s(t) is defined as

$$s(t) = x(t) + jH[x(t)] \tag{3.11}$$

where H[x(t)] is the Hilber transform of x(t). However, an alternative approach can be used to calculate the analytic signal using the frequency domain definition. The analytic signal s(t) can be evaluated by calculating the FFT of the time signal x(t), then setting the negative frequency spectrum to zero. The analytic signal can be obtained by evaluating the inverse FFT of the spectrum.

To simplify the computational effort, the WVD can be evaluated using a standard FFT algorithm. Adopting the convention that the sampling period is normalized to unity, it is necessary only to evaluate the WVD at time zero. Hence

$$W_x(0,f) = 2\sum_{i=-L}^{L} k(i)e^{-j4\pi fi} \tag{3.12}$$

where k (i) = s(i) s^* (i).

In order to avoid a repetition in the time domain WVD, a weighting function [28] was added to the time data before the evaluation process. Such a process may decrease

the resolution of the distribution, but it will eliminate the repetition of peaks in the time domain and the interpretation of the result is substantially easier.

3.3.2 Frequency Domain Analysis

The frequency spectrum is found by applying a Discrete Fourier Transform (DFT) on the time averaged signal x(t), such that the spectral components are

$$X(k) = T\sum_{t=0}^{N-1} x(t)\exp\left(\frac{-i2\pi ik}{N}\right) \tag{3.13}$$

where

x(t) = time domain signal

X(k)= frequency domain signal

T = sampling time interval

N = number of data points

The frequency components were examined in the frequency domain.

3.4. Discussions of Results

In order to examine the sensitivity of the system vibrations on gear tooth imperfections, the vibrations in gear systems due to various levels of gear wear were analyzed. The basic parameters used in this analysis are the magnitude and geometry of tooth profile deviation and the number of teeth involved.

The model of rotor-gear system used in this analysis is shown in Figure 15. The number of teeth in the gear model is 28. And the vibration analysis of this system under various levels of gear wear are given as follows.

First, surface pitting on a single tooth is used with the magnitude of the tip relief given in Table 3. The damaged tooth is the 12th tooth from the reference point on the gear.

Table III Magnitude of the Maximum Tip Relief

	case I			case II			case III		
1st tooth	1.e-5	2.e-5	4.e-5	4.e-5	4.e-5	4.e-5	4.e-5	4.e-5	4.e-5
2nd tooth	0	0	0	1.e-5	2.e-5	4.e-5	4.e-5	4.e-5	4.e-5
3rd tooth	0	0	0	0	0	0	1.e-5	2.e-5	4.e-5

Figure 16 - 18 show the WVD, the time signal (to the left of the WVD), and the frequency spectra (below the WVD) of the vibrations of the corresponding system. In Figure 16, a small cross pattern had developed in the WVD around the 154 degrees location which is the exact mesh time of the 12th tooth. A small phase shift is also detected in the time signal at that location. As the damage of the tooth becomes more severe (increase in the magnitude of the profile modification), the cross pattern is more obvious and the phase shift is getting more pronounced, as shown in Figure 17 and Figure 18.

Secondly, the dynamics involved with damage on two consecutive teeth (12th and 13th) was simulated. During this simulation the wear on the first tooth (12th) is kept constant while the amounts of wear on the second tooth (13th) increased similarly to that given in Table 3.

Figure 19 - 21 show the WVD, the time signal, and the frequency spectra of the vibrations of a corresponding system with damage on two consecutive teeth. When the damage on the tooth is increased, the cross pattern shown in the WVD is not as sharp as those with single tooth damaged. However the sideband components at the tooth pass frequency increase with increasing damage, as seen in the frequency spectra of Figures 19- 21. In addition, the once per revolution low frequency component increases along with its sidebands. As seen in the time domain, the phase shift becomes more pronounced als.o

Thirdly, the dynamics involved in three consecutive teeth (12th, 13th and 14th) were examined. In this case, the wear/damage in the first and the second teeth are kept constant, the amounts of wear in the third tooth increase similarly to those given in Table 3.

Figure 22 - 24 show the WVD, the time signal, and frequency spectra of the vibrations of the corresponding system with three consecutive teeth damaged. The frequency components near the tooth pass frequency show very small changes from those with two consecutive teeth damaged, Figure 19 -21. However, the frequency components near the shaft frequency has acquired a substantial increase with large sidebands.

Based on the above discussion, one can generalize the effects of single and multiple consecutive tooth damage on the vibrations of a gear transmission system. Using the WVD technique, 3-D image pattern recognition, procedures can be developed to identify various combinations and levels of tooth damage.

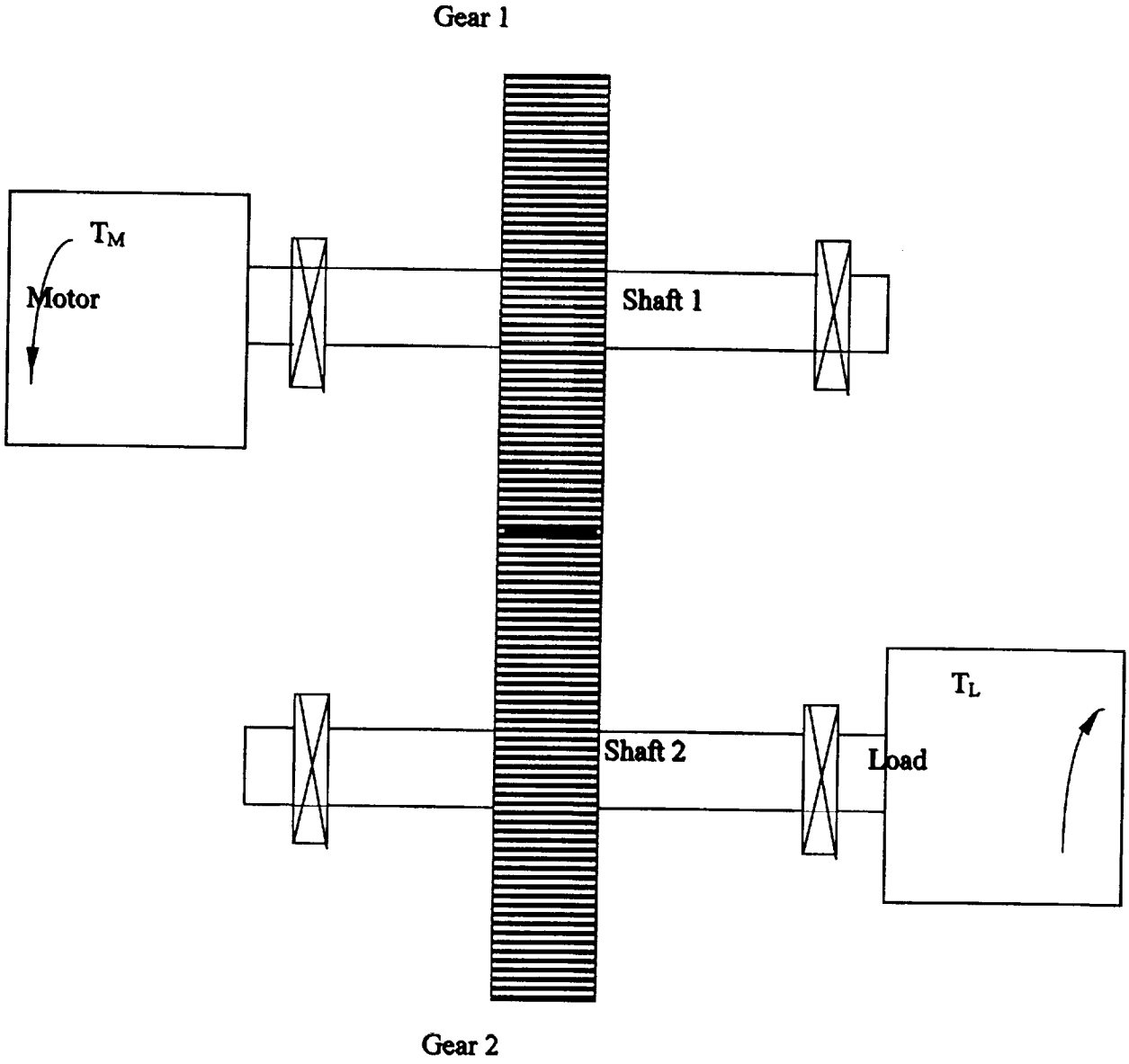


Figure 15. Gear transmission model

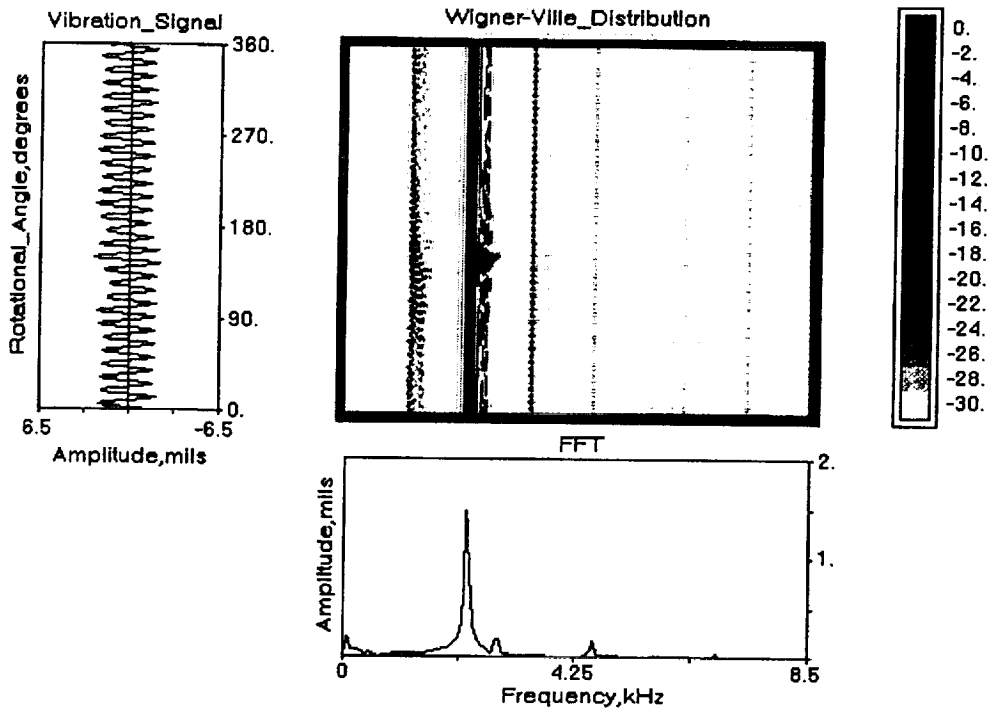


Figure 16. Simulated pinion vibration signature (only one tooth is damaged, tip relief is $1.0*10^{-5}$in).

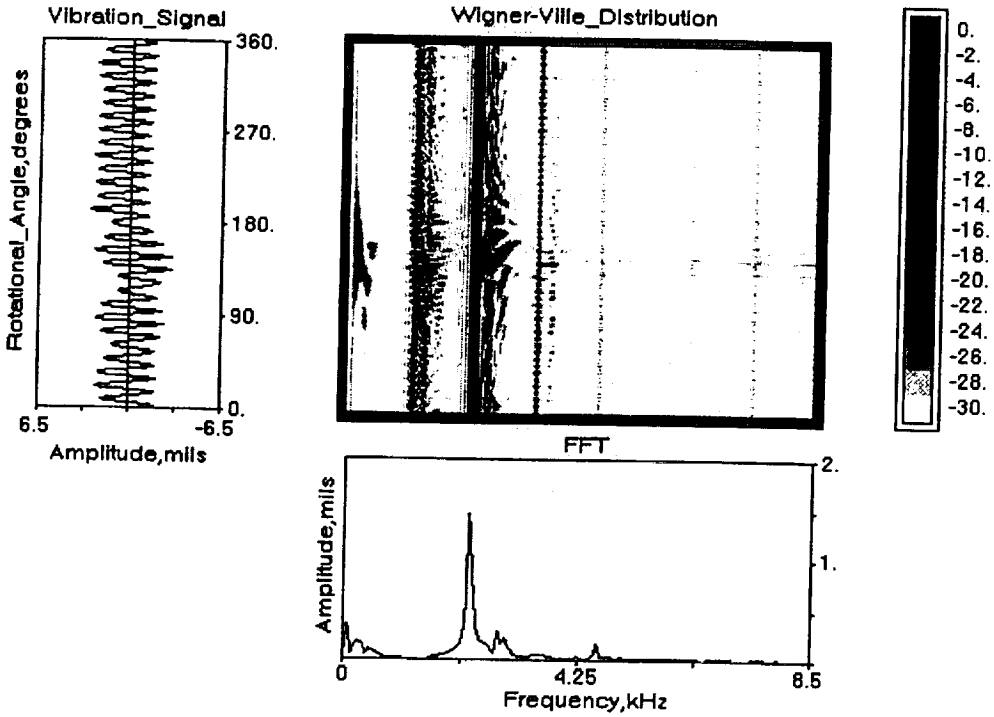


Figure 17. Simulated pinion vibration signature (only one tooth is damaged, tip relief is $2.0*10^{-5}$in).

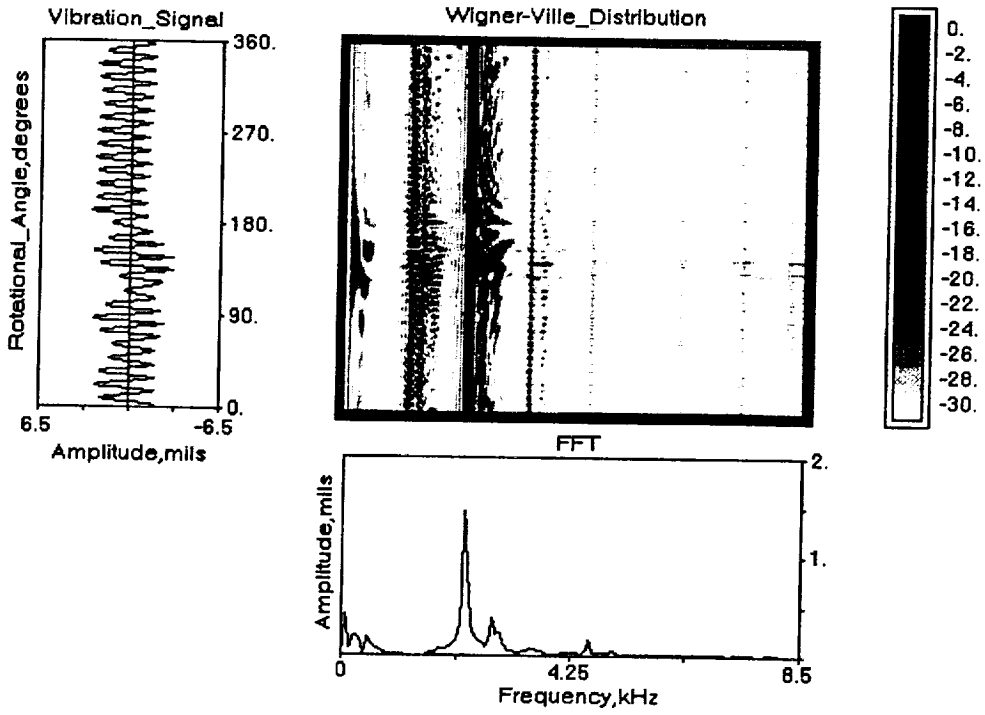


Figure 18. Simulated pinion vibration signature (only one tooth is damaged, tip relief is $4.0*10^{-5}$in).

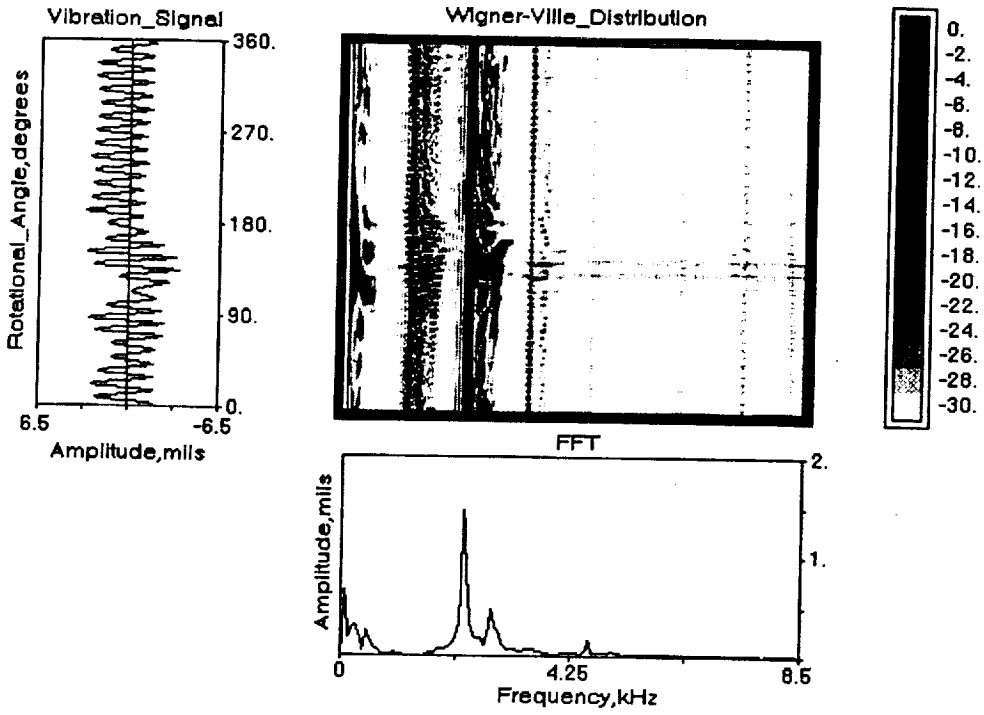


Figure 19. Simulated pinion vibration signature (two consecutive teeth are damaged, tip relief is $4.0*10^{-5}$in, $1.0*10^{-5}$ in respectively).

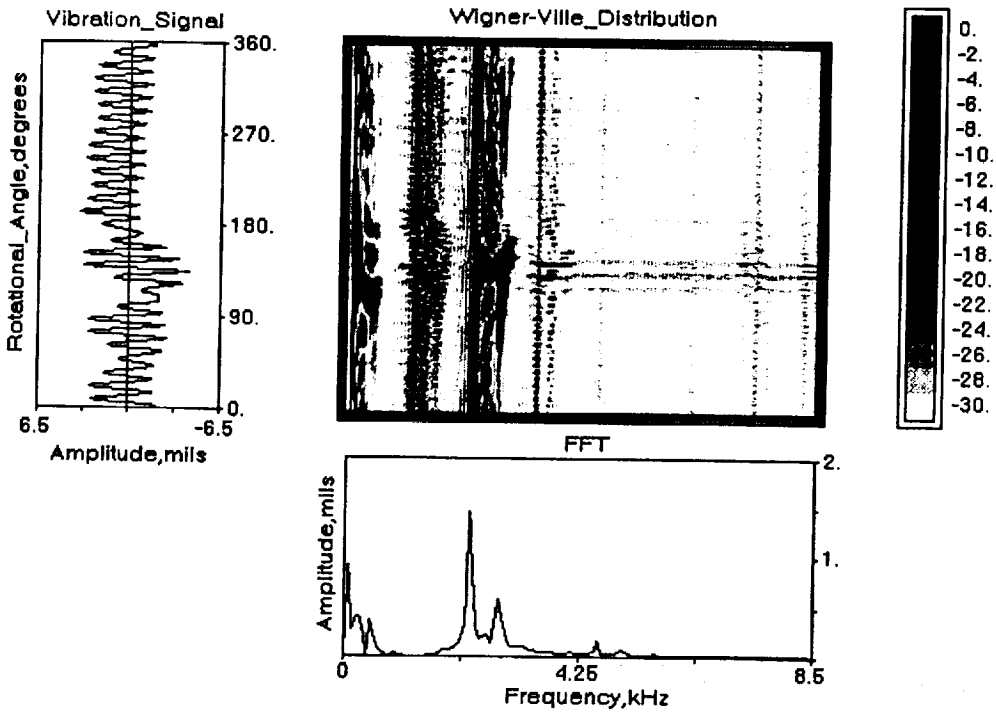


Figure 20. Simulated pinion vibration signature (two consecutive teeth are damaged, tip relief is $4.0*10^{-5}$in, $2.0*10^{-5}$ in respectively).

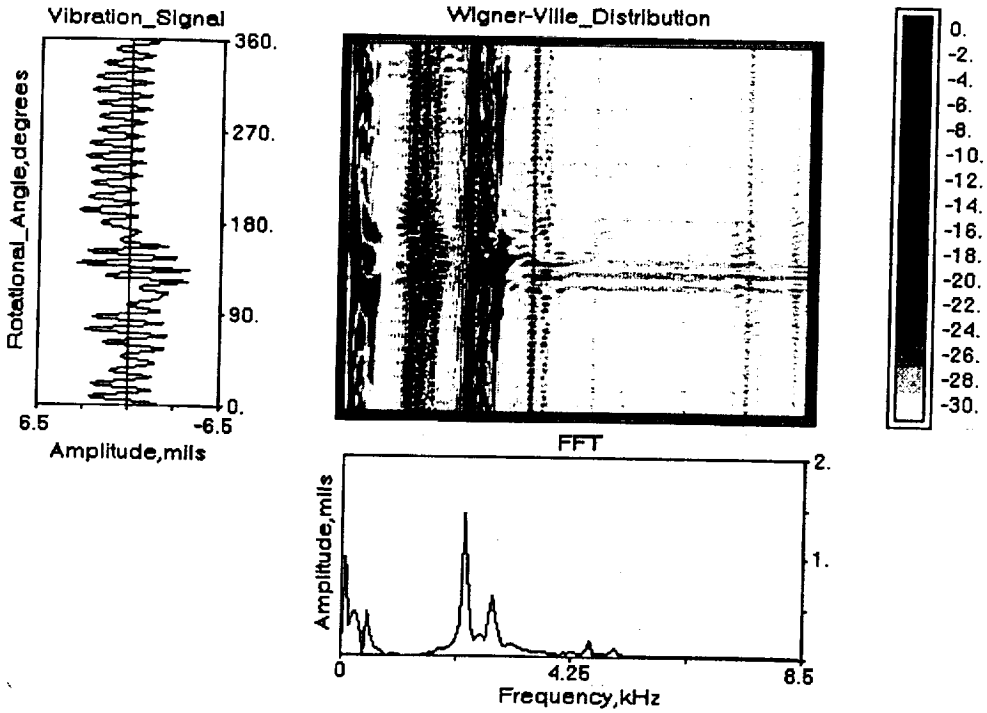


Figure 21. Simulated pinion vibration signature (two consecutive teeth are damaged, tip relief is $4.0*10^{-5}$in, $4.0*10^{-5}$ in respectively).

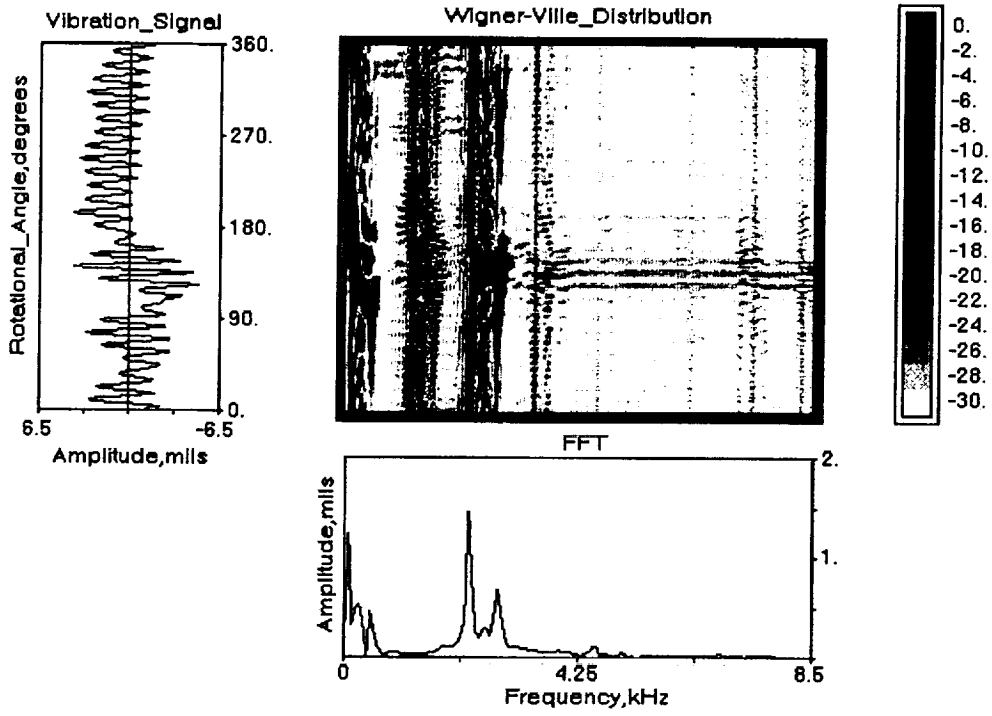


Figure 22. Simulated pinion vibration signature (three consecutive teeth are damaged, tip relief is $4.0*10^{-5}$in, $4.0*10^{-5}$ in, $1.0*10^{-5}$ in respectively).

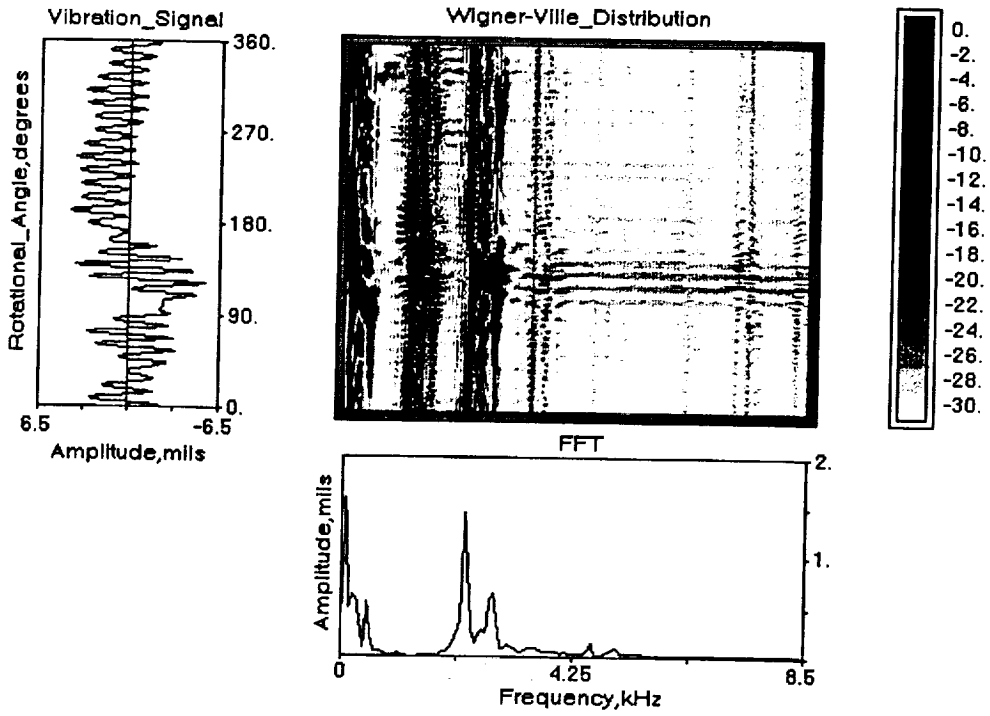


Figure 23. Simulated pinion vibration signature (three consecutive teeth are damaged, tip relief is $4.0*10^{-5}$in, $4.0*10^{-5}$ in, $2.0*10^{-5}$ in respectively).

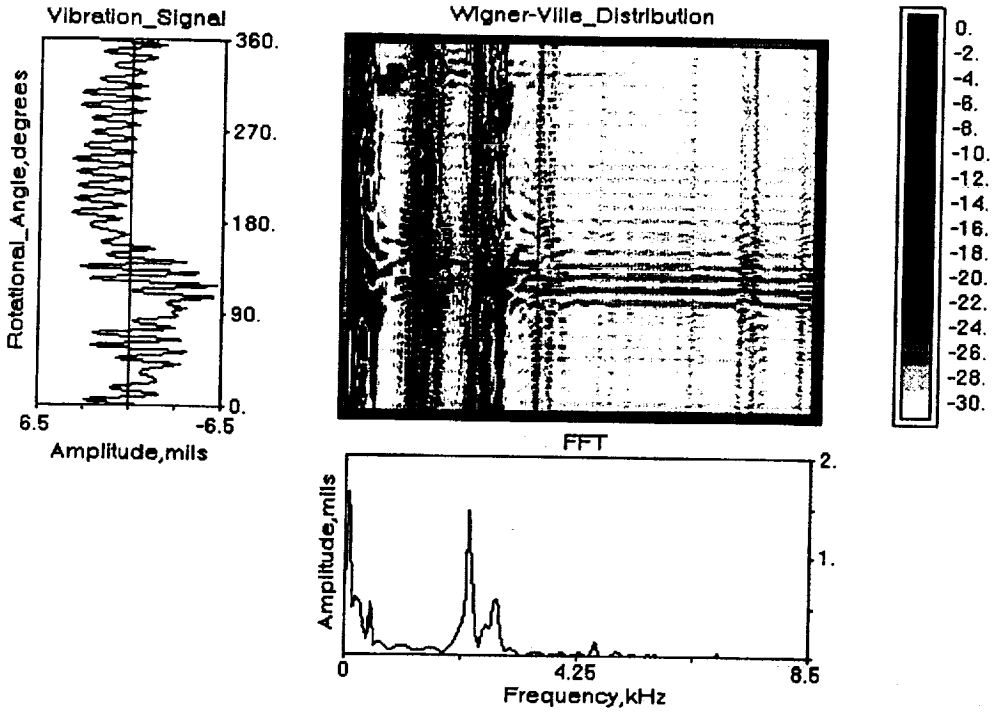


Figure 24. Simulated pinion vibration signature (three consecutive teeth are damaged, tip relief is $4.0*10^{-5}$in, $4.0*10^{-5}$ in, $4.0*10^{-5}$ in respectively).

3.5. Summary

A numerical procedure has been developed to simulate the vibration in a gear transmission system with effects of gear tooth damage due to wear and pitting. The work presented in this chapter can be summarized as follows:

1) A modal synthesis methodology has been used to simulate the dynamics of gear transmission systems. The computational efforts has been greatly reduced by modal transformation.
2) The gear mesh model developed to simulate the gear tooth damage due to wear and pitting can easily be incorporated into the global transmission system for dynamics predictions.
3) The Wigner-Villle distribution (WVD) method provides a comprehensive representation of the vibration signal. It was successfully used to verify the analytical model.
4) The WVD method provides detailed information. Hence, using the time averaging technique, frequency spectrum analysis, and the WVD, a signature analysis scheme can be developed to examine and characterize the vibration signal of the gear system.
5) A parametric study of the effects on the vibration signal due to various degrees of pitting and wear damage, could provide a comprehensive database for gear fault detection and damage estimation research.

CHAPTER 4

VIBRATION SIGNATURE ANALYSIS OF A FAULTED GEAR TRANSMISSION SYSTEM

4.1 Objective

The objective of this chapter is to examine and compare the three different approaches to detect gear wear and failure. The frequency domain analysis is based on the spectral display from a Fast Fourier Transform algorithm. The time domain analysis includes the time synchronous averaged signal, and the statistical based techniques FM0, FM4, NA4*, and NB4* applied to the time averaged signal. The joint time-frequency analysis uses the WVD on the vibration data with special windowing techniques. All of the analysis methods are applied to the vibration data of a failure from the spiral bevel gear fatigue test rig in the NASA Lewis Research Center. Results from the various methods are compared and general conclusions are drawn from the results.

4.2 Technical Approach

As discussed in the previous section, three major methodologies: A) the frequency domain approach, B) the time domain approach, and C) the joint time-frequency approach are used in this study. The following is a description of the three methodologies:

(A) Frequency Domain Techniques

The frequency spectrum is found by applying a discrete FFT on the time averaged signal x(t), such that the spectral components are

$$X(k) = T\sum_{i=0}^{N-1} x(t)\ \exp(\frac{-j2\pi ik}{N}) \tag{4.1}$$

where x(t)= time domain signal, X(k)= frequency domain signal, T= sampling time interval, N= number of data points. The frequency components are examined in the frequency domain and compared with those obtained at various stages of the fault development in the spiral bevel pinion.

(B) Time Domain Techniques

Four different time domain techniques for early detection of gear tooth damage are used in this study for evaluation and comparison. All of the time domain techniques are applied to the vibration signal after it has been time synchronously averaged. These techniques are: FM0, FM4, NA4*, and NB4*. These parameters are defined as follows:

FM0[46] is a course fault detection parameter that compares the maximum peak to peak amplitude to the sum of the mesh frequencies and its harmonics

$$FM0 = \frac{PP}{\sum_{i=1}^{n} A(f_i)} \tag{4.2}$$

where PP= maximum peak to peak amplitude in signal, $A(f_i)$= amplitude at mesh frequency and harmonics, n= total number of harmonics in frequency range, and FM4[46] is an isolated fault detection parameter, and is given by the normalized kurtosis, of the resulting difference signal as

$$FM4 = N\ \sum_{i=1}^{N}(d_i - \bar{d})^4 \Big/ \left[\sum_{i=1}^{N}(d_i - \bar{d})^2\right]^2 \tag{4.3}$$

where d(t)= A(t) - R(t), ^{-}d 2= mean value of d(t), A(t)= original time synchronous signal, R(t)= regular meshing components plus their first order side bands, and N = total number of data points in the time signal.

NA4*[45-48] is a general fault detection parameter with trending capabilities. A residual signal is constructed by removing regular meshing components from the time averaged signal. For NA4, the first order sidebands stay in the residual signal and the fourth statistical moment of the residual signal is then divided by the averaged variance of the residual signal, raised to the second power. The average variance is the mean value of the variance of all previous records in the run ensemble. This allows NA4 to compare the current gear vibration with the baseline of the system under nominal conditions. NA4 is given by the quasi-normalized kurtosis equation shown below:

$$\text{NA4*(M)} = \text{N} \sum_{i=1}^{N}(r_i - \bar{r})^4 \Big/ \left\{\frac{1}{M}\sum_{j=1}^{N}\left[\sum_{i=1}^{N}(r_{ij} - \bar{r}_j)^2\right]\right\}^2 \tag{4.4}$$

where r = residual signal, $\bar{r}$ 3= mean value of residual signal. N= total number of data points, in one time record, i= data point number in time record, j= time record number, and M=current time record number in run ensemble.

An enhancement to this parameter is given by NA4*, in which the value of the averaged variance is "locked" when the instantaneous variance exceeds a pre-determined value.[47] This provides NA4 with enhanced trending capabilities, in which the kurtosis of the current signal is compared to the variance of the locked baseline signal under nominal conditions.

NB4 is another parameter similar to NA4 that also uses the quasi-normalized kurtosis given in equation (4.4). The major difference is that instead of using a residual signal, NB4 uses the envelope of the signal banpassed about the mesh frequency. Again, as with NA4*, NB4* is an enhancement to the NB4 parameter, in which the value of the average variance is "locked" when the instantaneous variance exceeds a pre-determined value. The equation for NB4* is given below:

$$\text{NB4*(M)} = N \sum_{i=1}^{N} (s_i - \bar{s})^4 \Big/ \left\{ \frac{1}{M} \sum_{j=1}^{N} \left[\sum_{i=1}^{N} (s_{ij} - \bar{s}_j)^2 \right] \right\}^2 \qquad (4.5)$$

and

$$s(t) = \text{magnitude of } \{b(t) + i \{H[b(t)]\}\} \qquad (4.6)$$

where b(t)= time averaged signal bandpassed filtered about the meshing frequency, H[b(t)]=the Hilbert Transform of b(t), N= total number of data points in one time record, i= data point number in time record, j= time record number, and M= current time record number in run ensemble.

(C) Joint Time-Frequency Technique

To examine the vibration signal in a joint time-frequency domain, the Wigner-Ville method[14-18,41,49] is used in this study. While the FFT technique (eq. 4.1) can provide the spectral contents of the time signal, it cannot distinguish time phase change during a complete cycle of operation. In other words, it assumes that the time signals are repeatable for each time data acquisition window without considering the effects of any magnitude and phase changes during the sampling period. The Wigner-Ville distribution will provide an

interactive relationship between time and frequency during the period of the time data window. The WVD (Wigner-Ville Distribution) can be written as:

$$W(t, f) = \sum_{-\infty}^{\infty} x(t + \frac{\tau}{2})\, x^*(t - \frac{\tau}{2})\, e^{-j2\pi f \tau} d\tau \tag{4.7}$$

To allow sampling at the Nyquist rate and eliminate the concentration of energy around the frequency origin due to the cross product between negative and positive frequency,[14,15] the analytic signal was used in evaluating the WVD. The analytic signal s(t) is defined as

$$s(t) = x(t) + jH[x(t)] \tag{4.8}$$

where

H[x(t)] = the Hilbert transform of x(t) defined by:

$$H[x(t)] = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\xi)}{t-\xi}\, d\xi \tag{4.9}$$

However, an alternative approach can be used to calculate the analytic signal using the frequency domain definition. The analytic signal s(t) can be evaluated by calculating the FFT of the time signal x(t), then setting the negative frequency spectrum to zero. The analytic signal can be obtained by evaluating the inverse FFT of the spectrum. To simplify the computational effort, the WVD can be evaluated using a standard FFT algorithm. Adopting the convention that the sampling period is normalized to unity, equation (4.7) can be rewritten as

$$W_x(n, f) = 2 \sum_{i=-L}^{L} x(n+i)\, x^*(n-i)\, .\, e^{-j4\pi fi} \tag{4.10}$$

As for the continuous time case, it is necessary only to evaluate the WVD at time zero.

Hence

$$W_x(0, f) = 2\sum_{i=-L}^{L} k(i)\ e^{-j4\pi fi} \tag{4.11}$$

where k(i) = s(i) s* (-i). Equation (4.12) can be evaluated using the discrete FFT algorithm.

In order to avoid a repetition in the time domain WVD, a weighting function[18] is added to the time data before the evaluation process. Such a process may decrease the resolution of the distribution, but it will eliminate the repetition of peaks in the time domain and the interpretation of the result will be substantially easier.

4.3 Description of Experimental Procedure

The fatigue damage on the test pinion shown in Figures 25 to 32 was obtained using the spiral bevel gear fatigue test rig illustrated in figure 33, at the NASA Lewis Research Center. The primary purpose of this rig is to study the effects of gear tooth design, gear materials, and lubrication types on the fatigue strength of aircraft quality gears.[50] Because spiral bevel gears are used extensively in helicopter transmissions to transfer power between nonparallel intersecting shafts, the use of this fatigue rig for diagnostic studies is extremely practical. Vibration data from an accelerometer mounted on the pinion shaft bearing housing was captured using a personal computer with an analog to digital conversion board and anti-aliasing filter. The 12–tooth test pinion, and the 36–tooth gear have:, 0.5141 in pitch, 35 degree spiral angle, 1 in. face width, 90 degree shaft angle, and 22.5 degree pressure angle. The pinion transmits 720 hp at nominal speed of 14,400 rpm. The test rig was started and stopped several times for gear damage inspection. The test was ended at

17.79 operational hours when a broken portion of a tooth was found visually during one of the shutdowns.

4.4. Discussions of Results

A series of pictures showing the deterioration of the pinion teeth at various stages of the test are illustrated in figures 25 to 32. In figure 1 the initiation of a small pit on one of the pinion teeth during the first shutdown, at about five and a half hour into the test is shown. As the test progressed, the rig was shut down seven more times to examine the severity of the pitting and its relationship with the corresponding vibrations. Figures 26 to 28 show the increase of the damaged area at the pinion tooth as the elapsed time increased to 6.55, 8.55, and 10.03 hr respectively. Note that in figure 29, as the elapsed time increased to 12 hr, the damage of the pinion tooth increased to 75 percent of the tooth surface. At this stage, pitting also initiated on the adjacent tooth and continued to grow as the time increased to 14.53 hr, figure 30. At 16.16 hr, the damage has grown to three adjacent teeth as shown in figure 31. The test was terminated when a breakage is detected on one of the three heavily pitted teeth at 17.79 hr, as shown in figure 32.

Figure 34 depicts the running speed of the test rig during various stages of the experiment. Note that there is some fluctuations present in the running speed after each shutdown, with a magnitude of approximately 6 percent about the nominal pinion speed of 14,400 rpm. There is a sharp change in speed at approximately 8.75 hr. These variations in speed create a substantial effect on the vibration signal, which is amplified in the NA4 (fig. 37), NB4 (fig. 38), and the WVD (fig. 45) analysis.

Figure 25. Photograph of pinion damage at 5.50 hr.

Figure 26. Photograph of pinion damage at 6.55 hr.

Figure 27. Photograph of pinion damage at 8.55 hr.

Figure 28. Photograph of pinion damage at 10.03 hr.

Figure 29. Photograph of pinion damage at 12.03 hr.

Figure 30. Photograph of pinion damage at 14.53 hr.

Figure 31. Photograph of pinion damage at 16.16 hr.

Figure 32. Photograph of pinion damage at 17.79 hr.

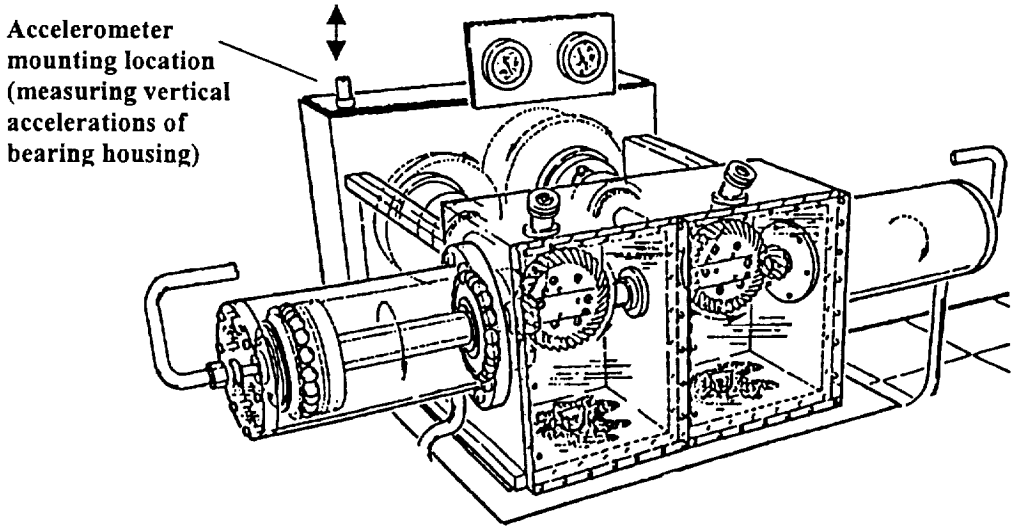


Figure 33. Spiral bevel gear rig at NASA Lewis Research Center.

Figure 35 shows the results of the FM0 analysis. As seen from the figure, FM0 shows only moderate changes as the damage starts and progresses. It does not provide any indication as the damages extends to the adjacent teeth, resulting in pinion tooth fracture. The majority of the variations in the FM0 parameter are most probably due to the speed changes experienced during the test.

Results from the FM4 parameter, as seen in figure 36, shows a possible reaction as the pitting started to occur, however, it does not provide any coherent indication of the severity of the pitting as the damage increased. In addition, it does not provide any information to distinguish the pitting of a single or multiple teeth.

Results from NA4 and NA4* are illustrated in figure 37. It is obvious from the figure that both NA4 and NA4* provide a very good indication of the pitting development on the pinion tooth. The magnitude of the parameter increases to a nondimensional value of 7 after shutdown #2 at 6.55 hr, and further to a value of 17 when the pitting covers 75 percent of the tooth surface at 8.5 hr. As expected, the "locked" denominator in NA4* provides a more robust indication as the pitting progresses.[45,47] Again, both parameters are very sensitive to the speed variations, especially after shutdown #3 and #6.

The NB4 and NB4* parameters, as shown in figure 38, show a very similar trend to those of NA4 and NA4*, with a more robust indication to the severity of the damage. However, both NA4 and NB4 did not provide any type of indication as the damage spread to other teeth, and finally as tooth fracture occurred.

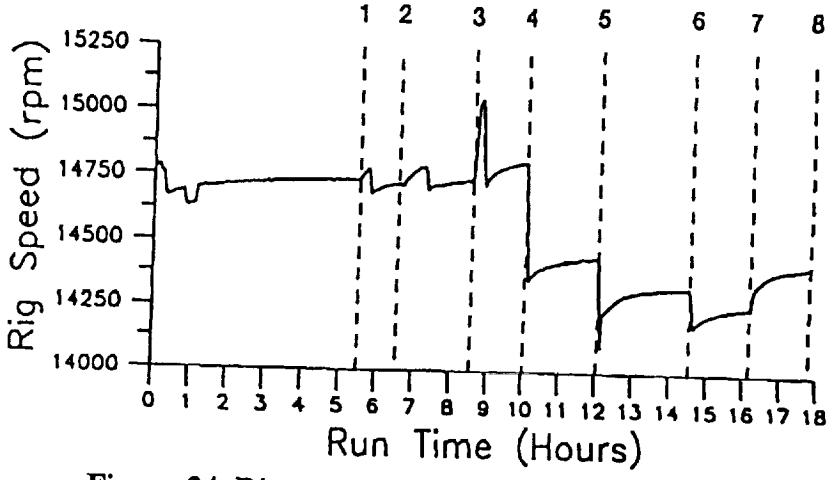


Figure 34. Rig operating speed at various run times.

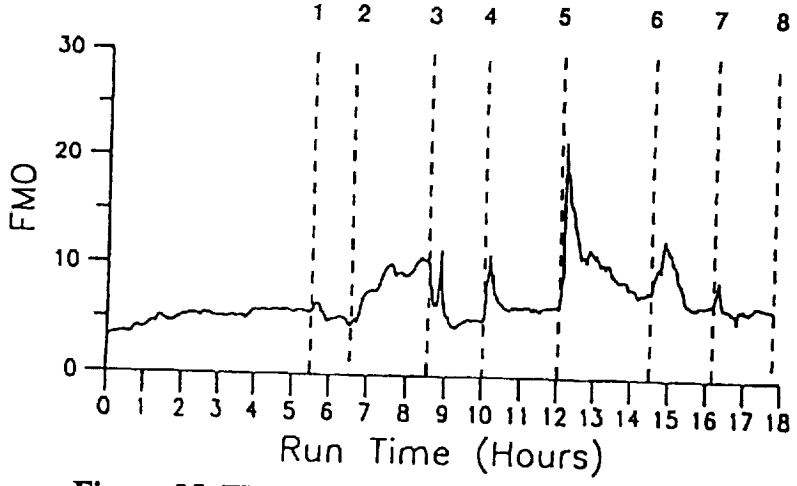


Figure 35. FM0 parameter at various run times.

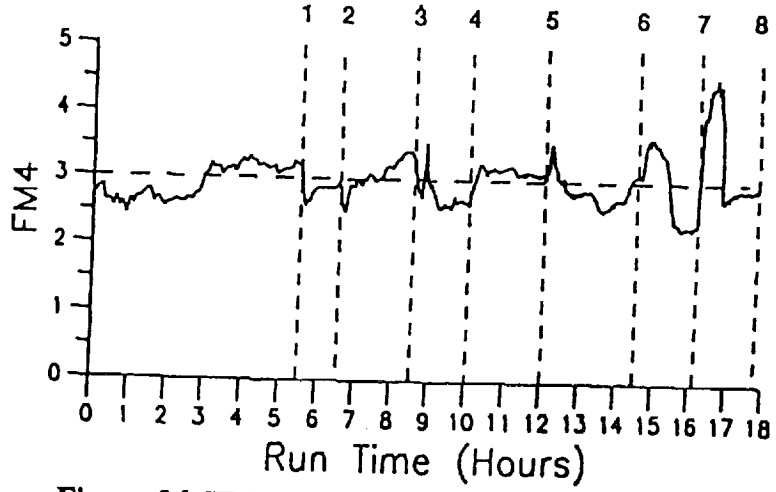


Figure 36. FM4 parameter at various run times.

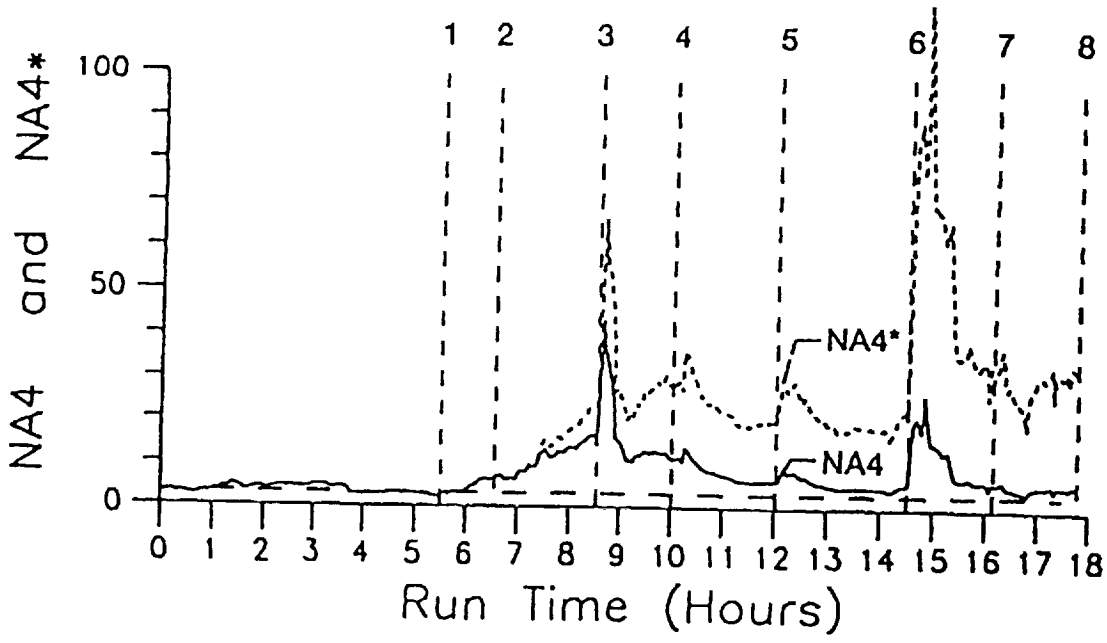


Figure 37. NA4 and NA4* at various run times.

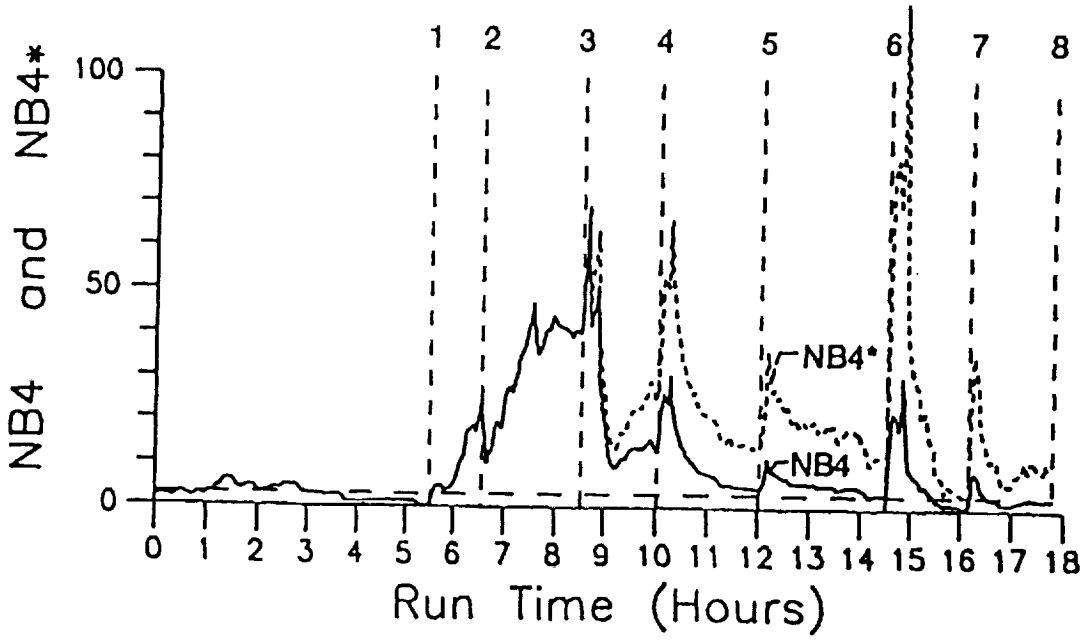


Figure 38. NB4 and NB4* at various run times.

Figure 39 shows a WVD (and corresponding intensity scale) and below the WVD the frequency spectrum (from FFT analysis) of a uniform sine wave signal shown at the left side of the figure. Note that only the frequency component of the input frequency is detected by both the frequency domain analysis and the joint time-frequency distribution. The WVD does not exhibit any changes during the 1 cycle (0 to 360 degrees) rotation of the shaft. When a short term amplitude and phase change is added to the system, as shown in figure 40, the frequency spectrum remains virtually unchanged. The WVD shows a dramatic change of the energy distribution pattern at the location where the change occurs. The lighter shades of the distribution display indicates a smaller vibration amplitude, which is shown by the time signal at the left side of the figure. Such an effect could be possibly caused by a chipped or cracked tooth. Figure 41 shows the effects of a short term amplitude increase in the time signal to simulate vibrations caused by gear tooth surface damage. Note that the frequency spectrum remains the same showing only the component of the exciting frequency while the WVD again provides a good indication of the amplitude increase by the widening of the shaded area to a diamond shape at the corresponding "damaged" tooth location. Figure 42 shows the effects of a time decaying short term amplitude and phase change signal. The WVD shows a half diamond shape of shaded area, similar to that of figure 40, at the location where the amplitude and phase changes are presented. As seen in figure 42, the frequency spectrum gives very little indication of the signal change.

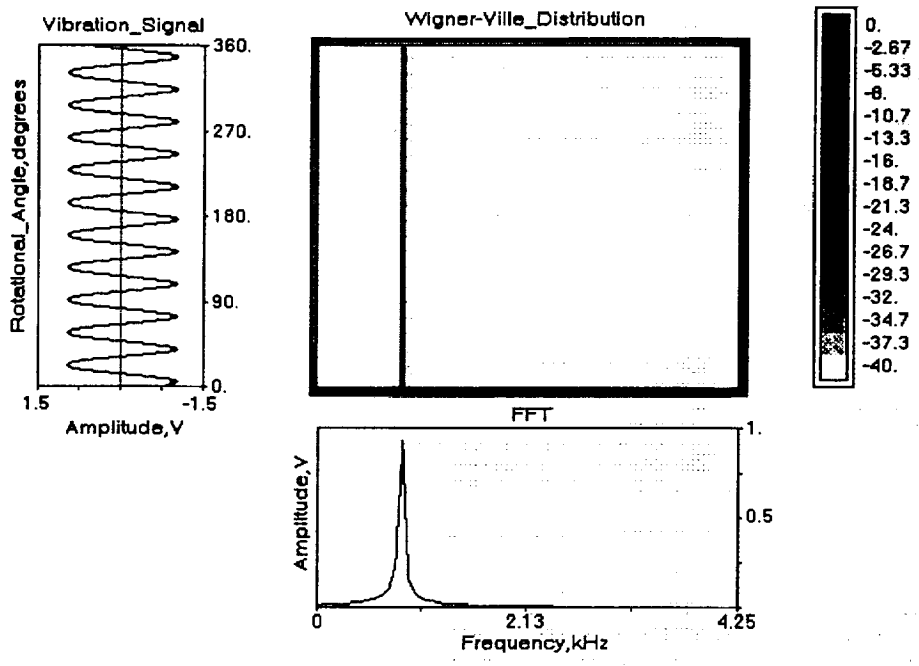


Figure 39. Example of WVD on a regular sine wave time signal.

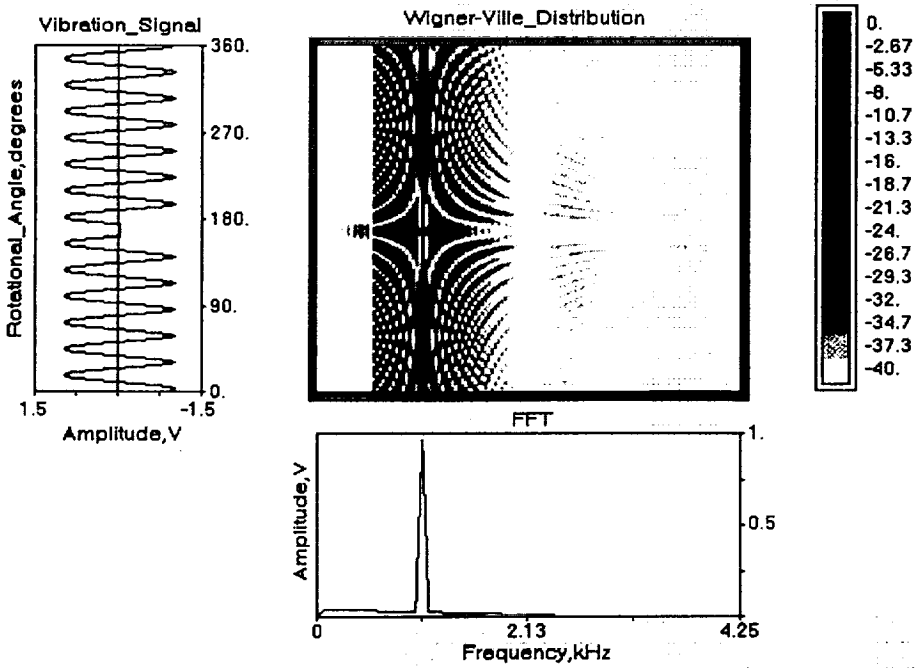


Figure 40. Example of WVD on a sine wave time signal with short term amplitude and phase change.

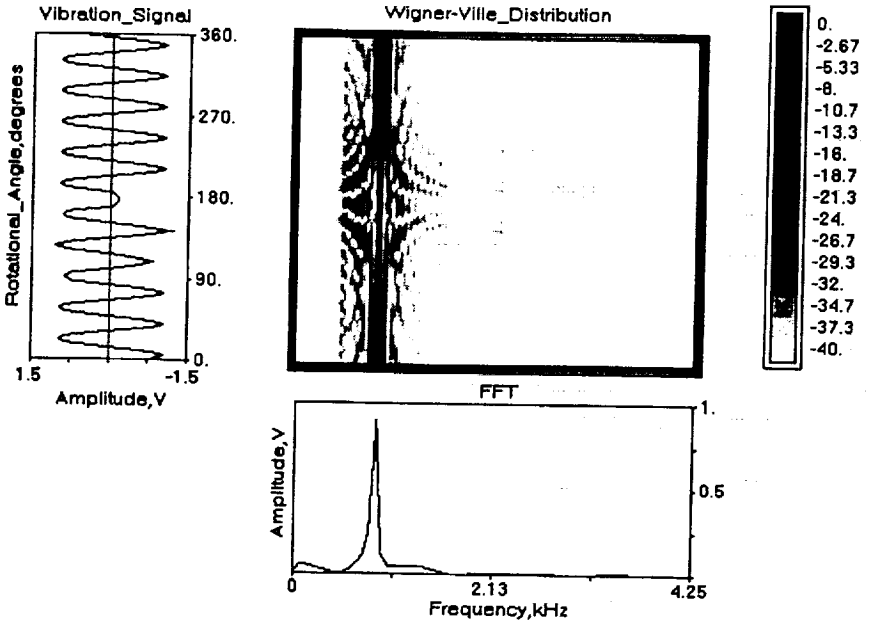


Figure 41. Example of WVD on short term amplitude increase in a signal.

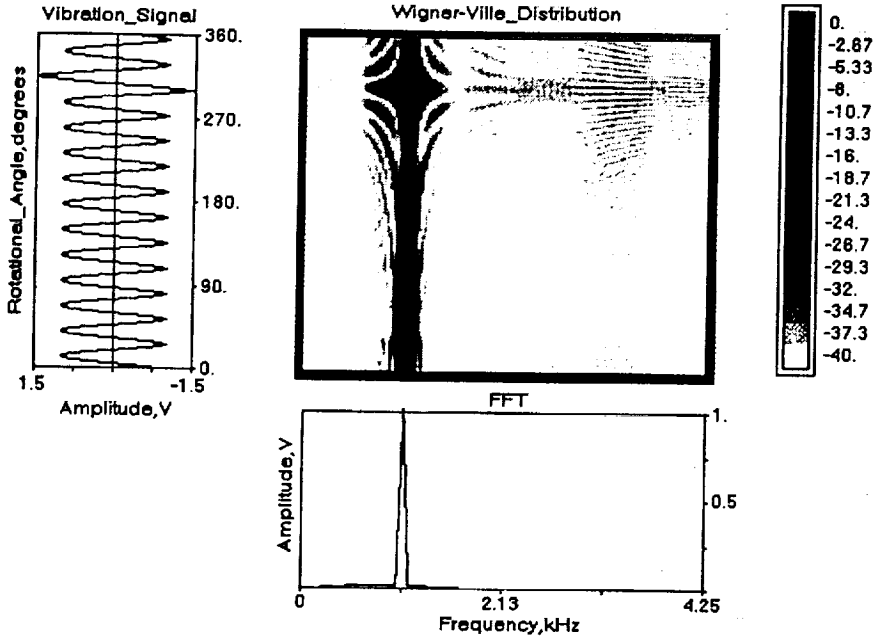


Figure 42. Example of WVD on short term amplitude and phase change in a signal.

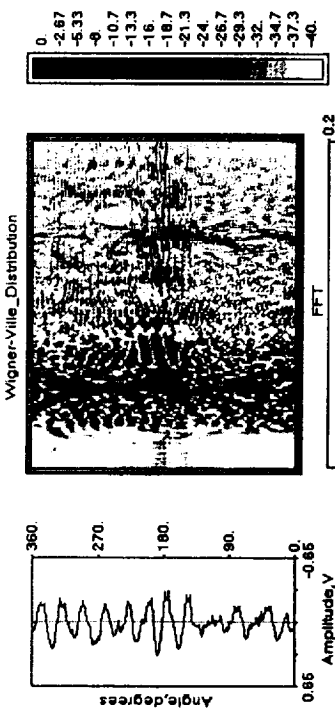


Figure 43. Application of WVD to spiral bevel gear vibration signal at 5.50 hr

Figure 44. Application of WVD to spiral bevel gear vibration signal at 6.56 hr

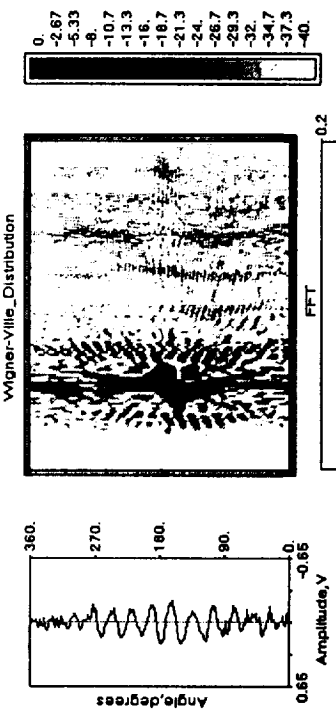


Figure 45. Application of WVD to spiral bevel gear vibration signal at 8.63 hr

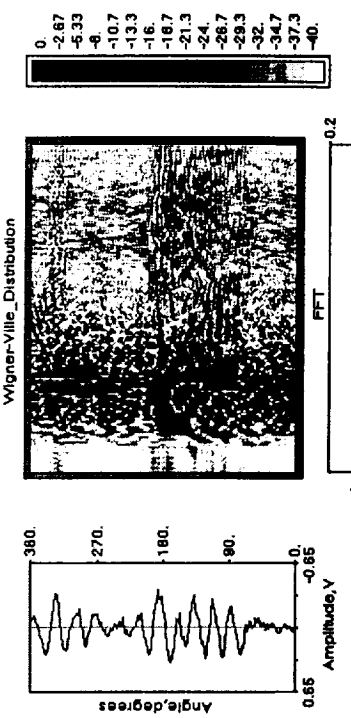


Figure 46. Application of WVD to spiral bevel gear vibration signal at 10.09 hr

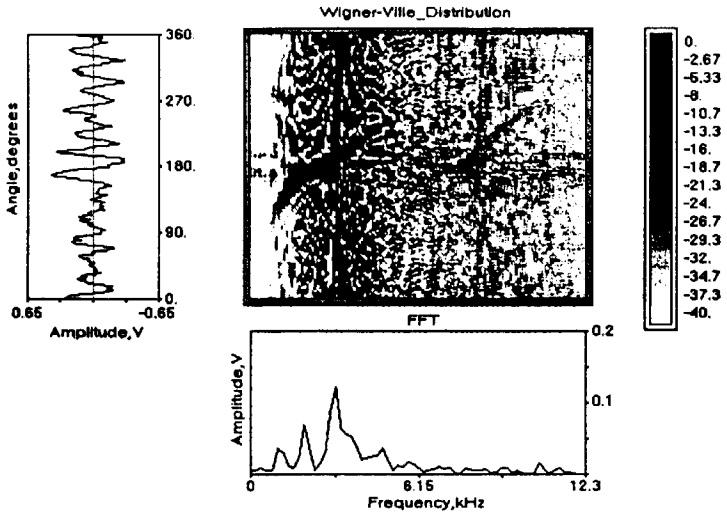


Figure 47. Application of WVD to spiral bevel gear vibration signal at 12.09 hr

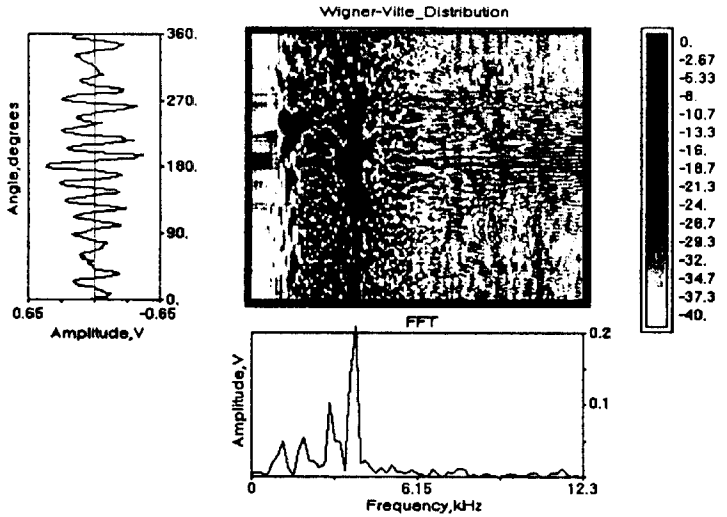


Figure 48. Application of WVD to spiral bevel gear vibration signal at 14.59 hr

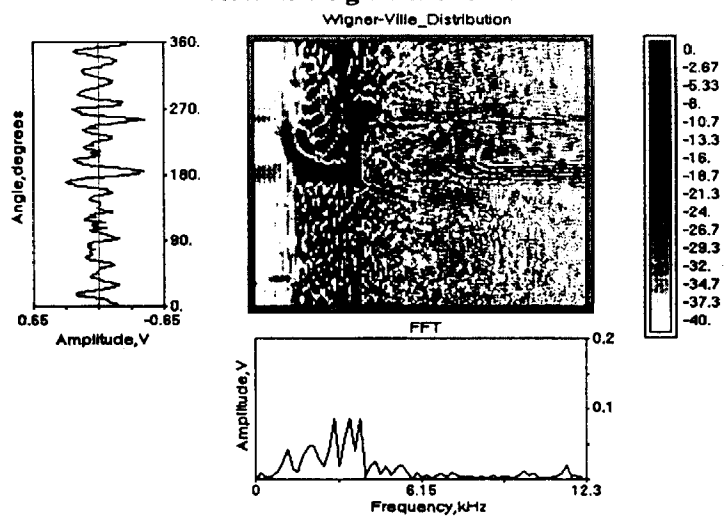


Figure 49. Application of WVD to spiral bevel gear vibration signal at 16.22 hr

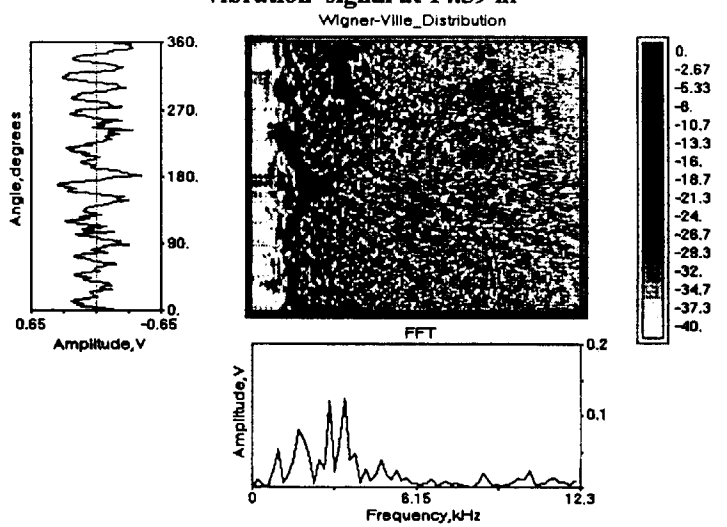


Figure 50. Application of WVD to spiral bevel gear vibration signal at 17.72 hr

Figures 43 to 50 show the WVD and frequency spectra of the spiral bevel pinion vibration at various stages of damage, corresponding to the photographs given in figures 25 to 32. Figure 43 shows the occurrence of the initial pitting at around 200 degrees from the triggering point of the gear, at 5.5 hr, of running time. At 6.55 hr, the pitting on the tooth surface progressed to a more noticeable stage, as shown in figure 26, the WVD pattern, figure 44, begins to adopt those of a short term amplitude and phase change as illustrated in figure 42. At this stage, a Cross pattern appeared in the joint time frequency domain (WVD) as the damaged gear tooth produced a change in the frequency components other than the mesh frequency. Due to the speed increase at 8.55 hr, the overall WVD amplitude increases substantially as shown in figure 45. At the running time of 10.03 hr, the corresponding WVD in figure 46 shows the initiation of a cross pattern as surface pitting in the damaged tooth becomes more pronounced which can also be evidenced from the photograph of the pinion gear in figure 28. This phenomenon is also evident in the frequency spectrum with the existence of sideband components. At 12.03 hr, when the pitting on the pinion tooth has extended to about 75 percent of the tooth surface as shown in figure 29, the WVD pattern exhibits a solid cross pattern extending over the mesh frequency and several of its adjacent sidebands. The high concentration in the WVD energy and the initiation of a second cross pattern at 14.53 hr, figure 48, shows the advancement of the pitting process on second tooth, as illustrated in figure 30. This is further confirmed by the large amplitude of sideband component(above mesh) in the frequency spectrum. At 16.16 hr, as seen in figure 49, the WVD pattern changes, showing more advanced damage pattern similar to the multiples of the decay of a single short term amplitude increase and phase change demonstrated in figure 42. Such phenomenon is due to the pitting of three

consecutive teeth in the pinion gear as shown in the picture given in figure 31. The frequency spectrum in figure 49 shows a substantial amplitudes increase in the sideband components. The discontinuity of the WVD at the mesh frequency, shown in figure 50, similar to the example shown in figure 16 due to the short term phase change, is probably the result of the instantaneous phase change caused by the fractured tooth, as illustrated in figure 32. The two cross patterns in the WVD is very distinct as the effects of pitting at the teeth adjacent to the fracture tooth become more pronounced. Note, also that, as given in figure 50, the amplitude of the sideband frequencies(above and below the mesh frequency) increase substantially.

4.5 Conclusions

Based on the results of the application of the various aforementioned methods, the following conclusions can be made:

1) The FM0 parameter shows only moderate changes as the damage starts and progresses. It also fails to indicate the fracture of the pinion tooth.
2) The FM4 parameter shows a possible reaction to the start of the pitting process, however, no coherent indication is provided for the growth and severity of the pit.
3) The NA4* and NB4* parameters show good reactions to the initial pitting damage and very nice indications for the growth and severity of the pitting damage. However their indications for the tooth fracture is somewhat unclear.
4) The WVD provides vital information concerning both the severity and the location of the pitting process in the gear system.

5) The fracture of the gear tooth and its exact location can be pinpointed using the WVD technique. However a machine vibration signature database is required to interpret the resulting WVD.

6) The occurrence and the severity of gear tooth failure can be reliably detected using a combination of the time averaging, the frequency analysis, and the WVD techniques.

CHAPTER 5

ANALYSIS OF THE EFFECTS OF SURFACE PITTING AND WEAR ON THE VIBRATIONS OF A GEAR TRANSMISSION SYSTEM

5.1 Objective

Vibration signature analysis methodologies are being developed to non-intrusively examine the health and wear of gear transmission systems. Using spectral analysis, the amplitude of the frequency spectrum of the measured vibration signal is calculated and displayed in a continuous manner. However, the spectral analysis technique is difficult to apply in a highly complex system where the large number of spectral lines often makes it difficult to detect significant changes in the spectrum. Another methodology is the joint time-frequency approach which applies the Wigner-Ville distribution (WVD) [14-16] on the time vibration signal of the system. Unlike the regular Fourier transform process, the WVD provides an instantaneous frequency spectrum of the system at any instant throughout the sampling period (while FFT provides a averaged frequency spectrum of the total sampling period). The spectral density of the fundamental exciting frequency and its sidebands change as the shaft rotates through a complete revolution. Some success has been achieved in applying the WVD concept in the health monitoring of gear transmission systems[16-18,26]. However, a complete vibration signature database is needed for development of an effective pattern recognition scheme. In order to populate such databases, the development of an accurate analytical procedure to predict vibrations in gear systems due to wear and fatigue failure is necessary.

The objective of this paper is to develop a comprehensive procedure to simulate and analyze the vibration in a gear transmission system with effects of surface pitting and wear of the

gear teeth under normal operating conditions. The effects of changes in magnitude and phase of the mesh stiffness at one particular tooth or a number of consecutive teeth were evaluated in order to simulate the effects of surface pitting and wear. The effects of these localized changes in the gear mesh were incorporated into each gear-rotor model for the dynamic simulation[11,51,52]. The dynamics of each gear-rotor system were coupled with each other through the gear mesh interacting forces. The coupling between the rotors and the casing structure were generated through the bearing support forces. The global vibrations of the system were evaluated by solving the transient dynamics of each rotor system simultaneously with the vibration of the casing. In order to minimize the computational effort, the number of degrees-of-freedom of the system were reduced by using a modal synthesis procedure[11,51]. The global transient dynamics of the overall transmission system were calculated in the modal coordinates where the modal accelerations were integrated numerically to obtain the velocities and displacements at each time step. An FFT procedure was used to transform the transient vibrations into the frequency domain for signature analysis. In addition, the Wigner-Ville distribution[14-18,26,53] was also used to examine the gear vibrations in the joint time-frequency domain for vibration pattern recognition. Experimental vibration results obtained from a gear fatigue test rig at NASA Lewis Research Center[48] were used to verify the analytical procedure.

5.2. Solution Procedures

5.2.1 Dynamics of the Gear-Shaft Configuration and the Gearbox System

The dynamics of the ith individual gear-shaft system can be evaluated through the equations of motion for the vibrations of a individual rotor-bearing-gear system as shown in Figure 51[11,51], given in matrix form, as

$$[M]\{\ddot{W}_i\} + [K_s]\{W_i\} = \{F_{bi}(t)\} + \{F_{gi}(t)\} + \{F_{ui}(t)\} \qquad (5.1)$$

where [M] and [K_s] are respectively the mass and shaft stiffness matrices of the rotor, $\{W_i\}$ is the general displacement vector of the ith rotor in the its local coordinate system, and, $\{F_{bi}(t)\}$, $\{F_{gi}(t)\}$, and $\{F_{ui}(t)\}$ are respectively the force vectors acting on the ith rotor system due to bearing forces, gear mesh interactions, and mass-imbalances. In this model, the dynamics between the gearbox and the rotor are coupled through the bearing forces, which are evaluated by the relative motion between the rotor and the gearbox. The interactions between the each individual rotor are coupled through the gear forces generated by the relative motion of the two mating gears at the mesh point.

The equations of motion of the gearbox with p rotor systems can be expressed as

$$[M_c]\{\ddot{W}_c\} + [K_c]\{W_c\} = \sum_{i=1}^{p} [T_{ci}]\{F_{bi}(t)\} \qquad (5.2)$$

where [T_{ci}] represents the coordinate transformation between the ith rotor and the gearbox.

5.2.2 Evaluation of Bearing Forces

The bearing forces $\{F_{bi}(t)\}$ for the ith rotor can be evaluated as

$$\{F_{bi}(t)\} = [\dot{C}_{bi}](\{\dot{W}_i\} - [T_{ic}]\{\dot{W}_{ci}\}) + [K_{bi}](\{W_i\} - [T_{ic}]\{W_{ci}\}) \quad (5.3)$$

where $[C_{bi}]$ and $[K_{bi}]$ are respectively the damping and stiffness of the bearing, $[T_{ic}]$ is the coordinate transformation matrix for the gearbox with respect to the ith rotor, and W_{ci} are the casing displacements at the rotor locations.

5.2.3 Evaluation of Gear Forces

The gear forces generated from the gear mesh interaction[54] can be written as

$$\{F_{gi}(t)\} = \{F_{ri}(t)\} + \{F_{ti}(t)\} \quad (5.4)$$

where $\{F_{ri}(t)\}$ is the vector containing the gear forces and moments resulting from the relative rotation between the two mating gears and $\{F_{ti}(t)\}$ is the vector containing gear forces and moments due to the translational motion between the two gears. the forces and the torsional moments due to relative rotation between the kth location of the ith rotor and the lth loation of the jth rotor, can be respectively expressed[54,10] as

$$\{F_{rik}(t)\} = [D_{gi}]\{K_{gij}\}((R_{ik}\theta_{ik}) - (\overline{R}_{jl}\overline{\theta}_{jl})) \quad (5.5)$$

$$\{M_{rik}(t)\} = R_i [D_{gi}]\{K_{gij}((R_{ik}\theta_{ik}) - (\overline{R}_{jl}\overline{\theta}_{jl})) \quad (5.6)$$

where R_{ik} and θ_{ik} are the radius and angular displacements in the ith-rotor gear localized gear rotational coordinates, $\overline{R}_{jl}$ 7 and $\overline{\theta}_{jl}$ 8 are radius and angular displacements of the lth location at the jth rotor in the ith-rotor localized gear rotational coordinates, $[K_{gij}]$ is the gear mesh stiffness matrix between the ith and jth rotors, and $[D_{gi}]$ is a diagonalized matrix transforming the

localized gear coordinates into the ith rotor coordinate system. In addition, the translational forces can be represented by

$$\{F_{tik}(t)\} = [D_{gi}]\{K_{gij}\}^T [D_{gi}]^t ([T_{ij}]\{W_{jl}\} - \{W_{ik}\}) \quad (5.7)$$

where $[T_{ij}]$ is the transformation matrix between the ith and jth rotor coordinates. For a rotor with one gear mesh, the gear force vector $\{F_{gi}(t)\}$ defined by equation (5.4) will have non-zero elements only at the gear location, i.e., the kth station. As the gear rotates, the stiffness of the gear mesh changes as the gear mesh progresses from single to multiple tooth contact. This cycling effect of the gear mesh stiffness is the main source for the mesh frequency excitation in the system.

5.2.4 Modal Synthesis Procedure

In order to calculate the transient and steady state dynamics of the system, the coupled rotor and casing equations of motion must be solved simultaneously. To minimize the computational effort, the modal transformation[11,51] procedure will be applied to reduce the degrees of freedom of the global equations of motion. Using m undamped mode shapes of the ith rotor system $[\phi_{i1}, \phi_{i2}, \phi_{i3}, \ldots, \phi_{im}]$ and m_c undamped mode shapes of the gearbox $[\phi_{c1}, \phi_{c1}, \phi_{c3}, \ldots, \phi_{cmc}]$, the rotor displacement for the ith rotor can be written as

$$\{W_i\} = \sum_{j=1}^{m} A_{ij} \{\phi_{ij}\}$$

(5.8)

or

$$\{W_i\} = [\phi_i] \{A_i\} \qquad (5.9)$$

and , similarly, the gearbox displacements as

$$\{W_c\} = [\phi_c] \{A_c\} \qquad (5.10)$$

where $\{A_i\}$ and $\{A_c\}$ are the modal time functions of the ith rotor and the gearbox respectively. Using the expansion in equation (8), the equations of motion for the ith rotor in equation (5.1) can be written as

$$[M] [\phi_i] \{\ddot{A}_i\} + [K_s] [\phi_i] \{\dot{A}_i\} = \{F_{bi}(t)\} + \{F_{gi}(t)\} + \{F_{ui}(t)\} \qquad (5.11)$$

Premultiplying by $[\phi_i]^T$ and using the orthogonality conditions of the mode shapes[11], the ith rotor equations of motion can be written as

$$\{\ddot{A}_i\} + [\_^2] \{A_i\} = \{\overline{F}_{bi}\} + \{\overline{F}_{gi}\} + \{\overline{F}_{ui}\} \qquad (5.12)$$

where $[\wedge^2]$ is the diagonal matrix of the squares of the natural frequencies of the system, and

$$\{\overline{F}_{bi}\} = [\phi_i]^T \{F_{bi}(t)\} \qquad (5.13)$$

$$\{\overline{F}_{gi}\} = [\phi_i]^T \{F_{gi}(t)\} \tag{5.14}$$

$$\{\overline{F}_{ui}\} = [\phi_i]^T \{F_{ui}(t)\} \tag{5.15}$$

For the gearbox system, a similar transformation is carried out as equation (2) can be written as

$$\{\ddot{A}_c\} + [\_^2_c]\{A_c\} = \{\overline{F}_{cb}\} \tag{5.16}$$

For a system of k rotors, equation (5.12) can be repeated k times and solved with the casing equation (5.16) simultaneously for the modal accelerations $\{A_i\}$ and $\{A_c\}$. A numerical integration scheme is used to integrate the accelerations to obtain velocities and displacements at each time step for transient calculations [11].

5.3. Signature Analysis of Vibration Signal

5.3.1 Frequency Domain Analysis

The frequency spectrum analysis is used by applying a discrete Fourier Transform on the average time signal x(t) such that the spectral components are

$$X(k) = T \sum_{i=0}^{N-1} x(t) \exp(\frac{-j2\pi ik}{N}) \tag{5.17}$$

where x(t) is the time averaged of the vibration signal W(t) and T is the sampling interval. The frequency components are examined in the frequency domain and compared with those obtained at various stages of the fault development in the experimental gear test rig.

3.2 Joint Time-Frequency Analysis : The Wigner-Ville Distribution

To examine the vibration signal in a joint time-frequency domain, the Wigner-Ville method[14-16] is used in this study. The Wigner-Ville distribution will provide an inter-domain relationship between time and frequency during the period of the time data window. The WVD (Wigner-Ville Distribution) can be written as:

$$WV(t,f) = \int_{-\infty}^{\infty} x(t + \frac{\tau}{2})\, x^{*}(t - \frac{\tau}{2})\, e^{-j2\pi f\tau}\, d\tau \qquad (5.18)$$

or in a discrete form as

$$WV_x(nT,f) = 2T \sum_{i=-L}^{L} x(nT + iT)\, x^{*}(nT - iT).\ e^{-j4\pi fi\tau} \qquad (5.19)$$

where WV(t, f) is the Wigner-Ville distribution in both the time domain t and the frequency domain f. To allow sampling at the Nyquist rate and eliminate the concentration of energy around the frequency origin due to the cross product between negative and positive frequency[14-16], the analytic signal was used in evaluating the WVD. The analytic signal s(t) is defined as

$$s(t) = x(t) + jH[x(t)] \qquad (5.20)$$

Where H[x(t)] is the Hilbert transform of x(t) defined by :

$$H[x(t)] = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\xi)}{t-\xi} d\xi \qquad (5.21)$$

However, an alternative approach can be used to calculate the analytic signal by the frequency domain definition. The analytic signal s(t) can be evaluated by calculating the FFT of the time signal x(t), then setting the negative frequency spectrum to zero. The analyticsignal can then be obtained by evaluating the inverse FFT of the spectrum. To simplify the computational effort, the WVD can be evaluated using a standard FFT algorithm. Adopting the convention that the sampling period is normalized to unity, Equation (5.19) can be rewritten as

$$WV_x(n,f) = 2 \sum_{i=-L}^{L} x(n+i)\, x^*(n-i).\, e^{-j4\pi fi} \qquad (5.22)$$

As for the continuous time case, it is necessary only to evaluate the WVD at time zero. Hence

$$WV_x(0,f) = 2 \sum_{i=-L}^{L} k(i)\, e^{-j4\pi fi} \qquad (5.23)$$

where k(i) = s(i) s* (-i). Equation (23) can be evaluated using the discrete FFT algorithm.

In order to avoid a repetition in the time domain WVD, a weighting function[26] is added to the time data before the evaluation process. Such process may decrease the resolution of the distribution, but it will eliminate the repetition of peaks in the time domain, and, thus the interpretation of the result will be substantially easier.

5.4. Description of Experimental Study

The experiment was performed on the spiral bevel gear fatigue test rig [48], as illustrated in Figure 52, at the NASA Lewis Research Center. The primary purpose of this rig is to study the effects of gear tooth design, gear materials, and lubricants on the fatigue strength of aircraft

quality gears. Because spiral bevel gears are used extensively in helicopter transmissions to transfer power between nonparallel intersecting shafts, the use of this fatigue rig for diagnostic studies is practical. Vibration data from an accelerometer mounted on the pinion shaft bearing housing was captured by an analog to digital conversion board. The 12-tooth test pinion, and the 36-tooth gear have a 35 degree spiral angle, a 1 in. face with, a 90 degree shaft angle, and 22.5 degree pressure angle. The pinion transmits 720 hp at a nominal speed of 14,400 rpm. The test rig was stopped several times during the test for gear damage inspection. The test was concluded at 17.8 operational hours when a breaken tooth was detected visually during one of the shutdowns.

Pictures of tooth damage on the pinion at various stages in the test are shown in Figure 53. At the first rig shut-down, at about 5.5 hours into the test, a small pit was observed on one of the teeth on the test pinion, as illustrated in Figure 53A. The test was stopped again at approximately 12 hours and the pitted area spread to cover approximately 75% of the face of the pinion tooth, as seen in Figure 53B. In addition, pitting started to appear on the adjacent teeth. Figure 53B shows the pinion at the end of the test, at 17.8 hours. It was found that one of the three heavily pitted pinion teeth had experienced a tooth breakage, losing one third of the tooth, as shown in the figure.

5.5. Discussions of Results

To study the effects of gear tooth pitting and wear on the dynamics of the rotor system, the numerical simulation procedure described above was used to model the vibrations of the pinion gear in the test rig. During the experimental study, vertical direction vibration signals

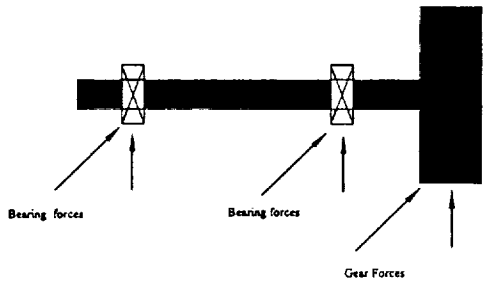


Figure 51. Schematic of the rotor-gear bearing system.

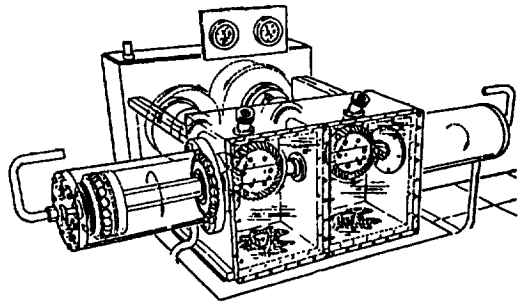


Figure 52. Spiral bevel gear rig at NASA Lewis Research Center.

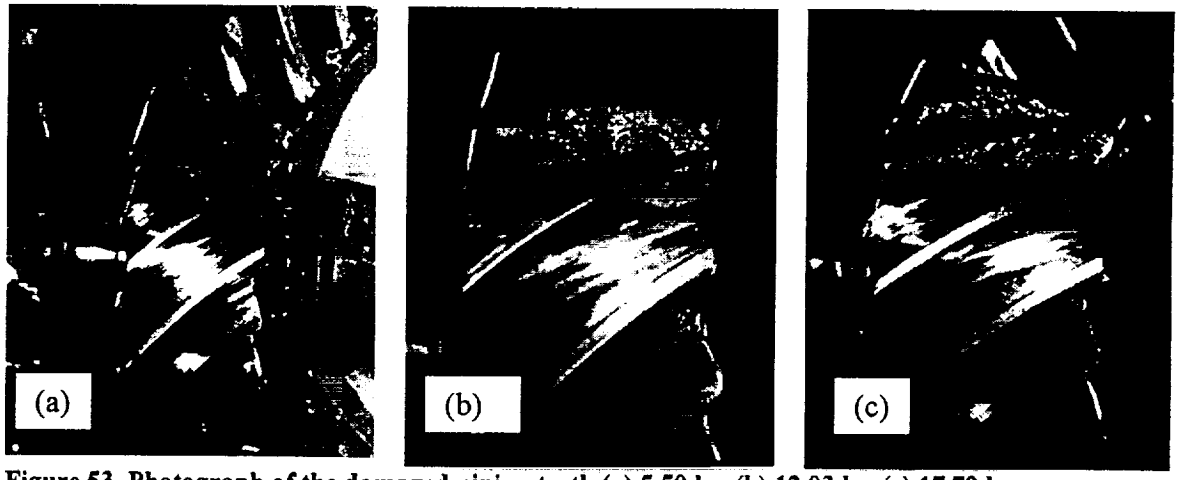


Figure 53. Photograph of the damaged pinion teeth (a) 5.50 hr. (b) 12.03 hr. (c) 17.79 hr.

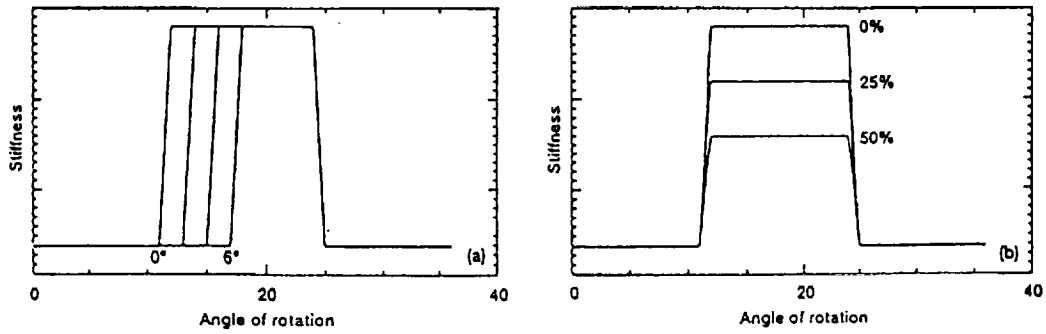


Figure 54. Stiffness changes in the gear mesh model.

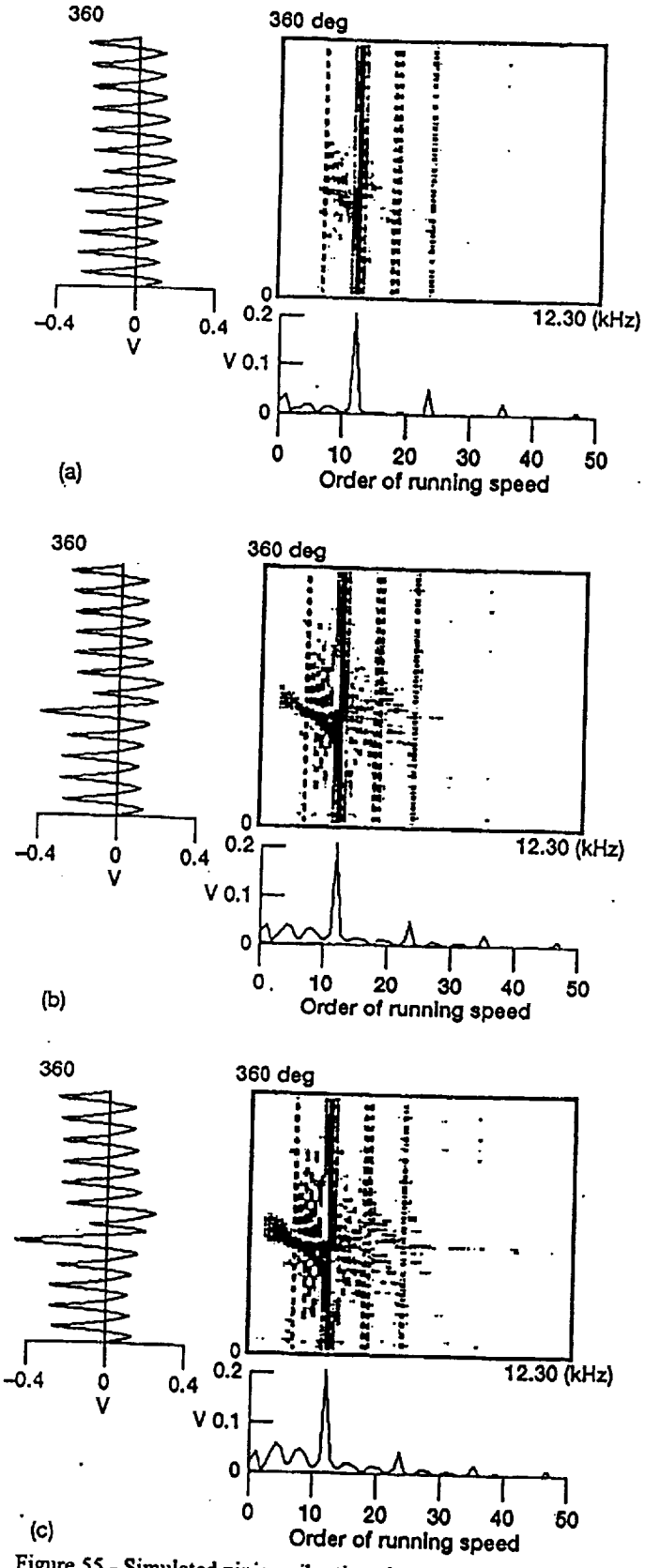


Figure 55 - Simulated pinion vibration signature due to phase change in the gear mesh stiffness (a) 1.5^0. (b) 3.0^0. (c) 4.5^0.

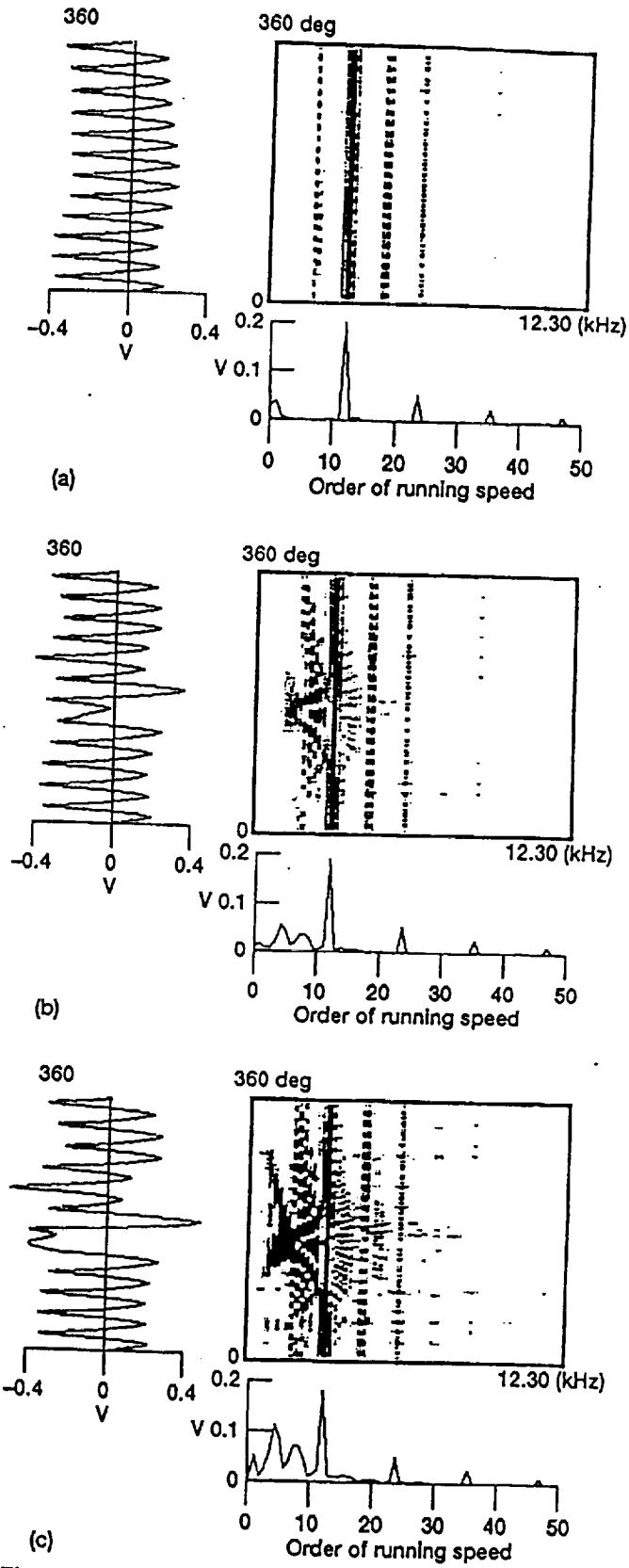


Figure 56 - Simulated pinion vibration signature due to amplitude change in the gear mesh stiffness (a) 0% reduction, (b) 25% reduction, (c) 50% reduction

from the pinion gear are time synchronously averaged for spectral analysis and analysis using the joint time-frequency distribution(WVD). In order to perform an accurate comparison, the averaged time signal from the vertical vibration of the pinion gear is also generated using the numerical model. During these simulations, approximate gear mesh stiffness models are developed to simulated the effects of wear and pitting of the pinion tooth on the dynamics of the system.

As it has been established, the changes due to gear tooth wear or failure can be represented by the amplitude and phase changes in vibration, which, in turn, can be represented by magnitude and phase changes in mesh stiffness[17,18]. To demonstrate the effects of mesh stiffness change on gear vibration, the variation of the mesh stiffness model used for this study is given in Fig 54. The "undamaged" configuration of the mesh stiffness is given by 0 degree phase change(Fig. 54A), and 0% amplitude reduction(Fig. 54B).During the wear and pitting process, two types of stiffness changes are examined, i.e., the phase changes, shown in Figure 54A, and the amplitude changes, shown in Figure 54B. Figures 55 and 56 show the time, frequency, and joint time-frequency analysis(WVD) of the pinion gear vibration signals with the approximated changes in gear mesh stiffness.

Figure 55 shows the effects of mesh stiffness phase changes in the WVD representation of the predicted vibration signal. As seen in Figure 55, a phase change in the mesh stiffness at the 6th tooth of the 12-teeth pinion resulted in a temporary increase of amplitude and phase of the pinion vibration time signal during the 6th tooth pass location. As the phase shift in the mesh stiffness increases, from 1.5 degrees to 4.5 degrees, the changes in amplitude and phase in the vibration signal become more pronounced. In the frequency spectra, this change in mesh stiffness

will result in the increase of the amplitude in the sideband frequencies. However, as discussed earlier, although the frequency spectrum provides good indications of the existence of the non-synchronous components, it can not distinguish the time locations of their occurrences. The joint time-frequency analysis using WVD, as indicated by the scale of shades given in Figure 55, shows the existence of various frequency components as the pinion rotates through a complete revolution of 360 degrees. Note that, in this case, the WVD shows a continuous excitation of the mesh frequency (12 x rotational speed) throughout the complete 360-degree revolution while subsynchronous components of 8 times, 4 times, and 1 times rotational speed are occurring at the 6th and 7th tooth pass locations.

Figure 56 shows the effects of reduction in mesh stiffness at the 6th gear location. With the reduction of mesh stiffness, a substantial change in the vibration at the 6th tooth pass location (150 - 180 degrees) is observed. Note that in the case of 50% stiffness reduction, the time vibration amplitude at the 6th tooth pass location(at approximately the 180 degree location for the 12 teeth pinion) almost vanishes and a much larger amplitude at the 7th tooth pass location is generated. In addition, the vibration amplitudes of the 8th and 9th tooth pass locations are reduced. These reductions in vibration amplitudes at mesh frequency resulted in a much higher sub-synchronous components in the frequency spectrum as shown in Figure 56C. The WVD shows a distinct type of cross pattern at the intersection of the mesh frequency and the 6th tooth pass location with a continuous mesh frequency component throughout the complete pinion revolution.

Figure 57 shows the pinion gear vibration signature analysis of the experimental time signal acquired at A) 12 hours when one tooth is severely pitted (Figure 53B), and B) 17.8

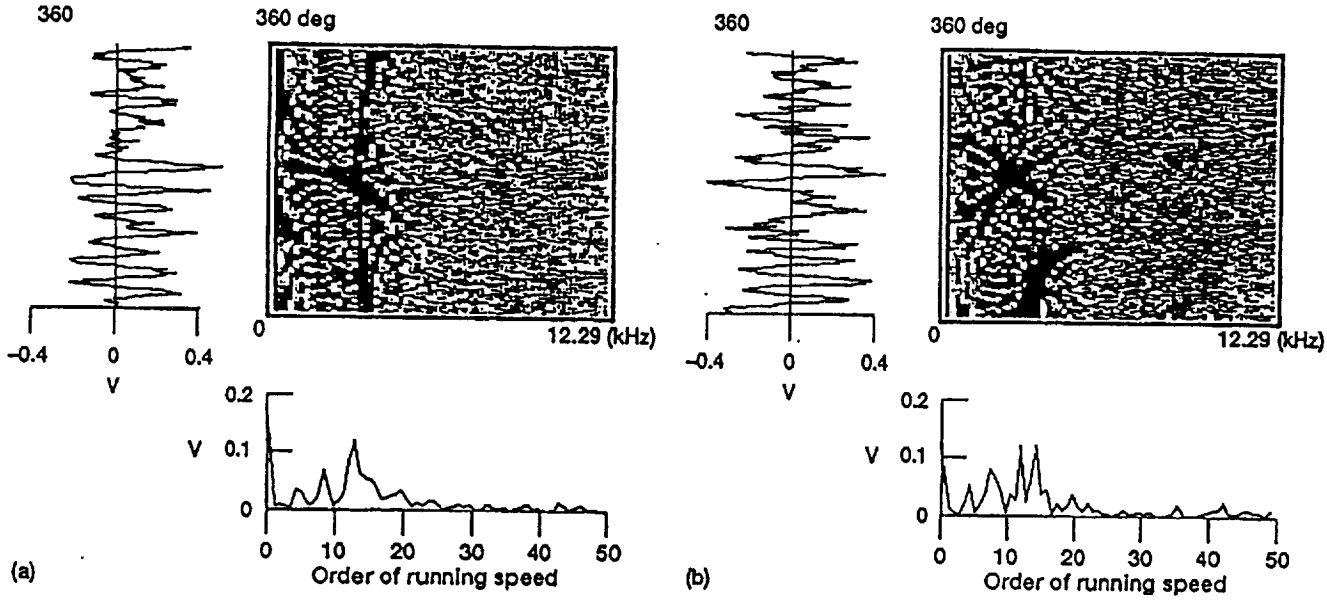


Figure 57 - Experimental pinion vibration due to damage on pinion teeth due to wear and pitting (a) single tooth (12hr), (b) three teeth (17.8hr)

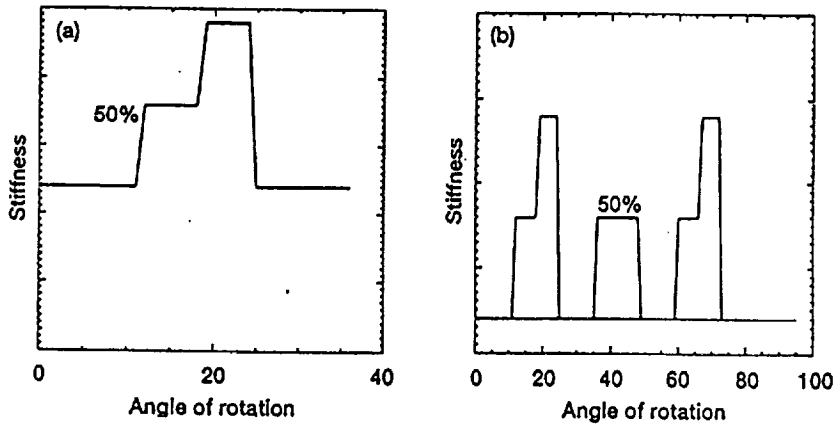


Figure 58 - Gear mesh stiffness model to simulate damages in pinion gear teeth

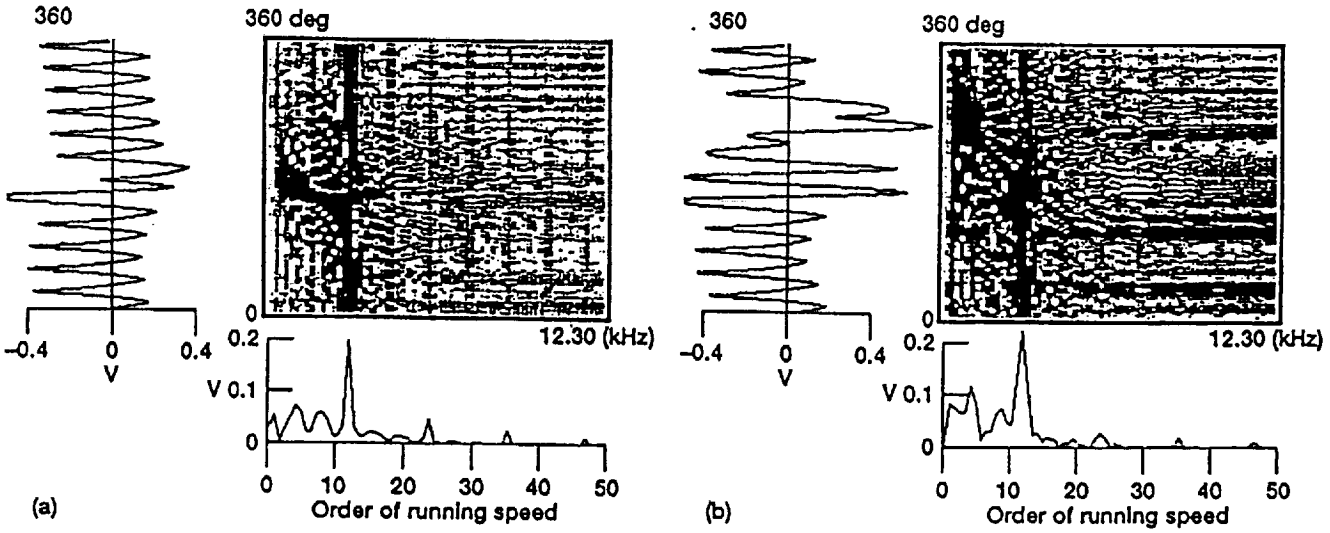


Figure 59 - Numerically simulated pinion vibration due to damage on pinion teeth due to wear and pitting (a) Single tooth, (b) Three teeth.

hours, when three consecutive teeth are pitted and one has a tooth fracture (Figure 53C). To numerically simulate these phenomena, two gear mesh stiffness models, as shown in Figure 58A and 58B, which include a combination of phase shift and amplitude change, are introduced. Figure 58A represents the gear mesh stiffness for a heavily pitted tooth during a single tooth pass. The mesh stiffness is simulated by a 50% loss in stiffness at approximately first 20% of contacting period. Figure 58B represent the mesh stiffness, for three tooth pass period, consisting of one broken tooth with two heavily pitted teeth at the adjacent sides. Note that the stiffness of the middle (broken) tooth is simulated by a 50% loss of stiffness at the first 40% of its contacting period while the other adjacent(pitted) teeth are simulated by the stiffness reduction similar to that of the single tooth case as shown in Fig. 58A. Additional frictional effects are also added into the model to simulate the roughness of the tooth surface due to pitting. The simulated vibration signature of the pinion gear is given in Figure 59. Comparing Figures 59A and 57a, for the single tooth damage case at 12 hours, one may notice the similarities between both the frequency spectra and the WVD display. Some of the unevenness in the experimental time signal is mainly due to the modulation of frequencies due to other excitations in the test rig which are not numerically modeled. For the tooth break-off case at 17.8 hours, Figures 57B and 59B, both the numerical and the experimental WVD display a large cross pattern at the 6th tooth pass location due to tooth break-off. However, some discrepancies have been detected between the experimental and the numerical time signal at the 4th and 5th tooth pass locations. The experimental time signal consists of some higher frequency, smaller amplitude vibration modulation, which are not being numerically simulated. This additional modulated signal resulted in the excitation of the 14 times rotational speed component, as shown in the frequency

spectrum in Figure 57B, and, also, in turn, is responsible for the small differences created in the WVD.

5.6. Summary and Conclusions

A numerical procedure has been developed to simulate the dynamics of gear transmission systems with the effects of gear tooth damage due to wear and pitting. The work presented in this paper can be summarized as follows:

1) A modal synthesis methodology has been developed to simulate the dynamics of gear transmission systems. While the computational efforts has been greatly reduced by modal transformation, the numerical results generated maintains good accuracy.

2) Gear tooth damage due to wear and pitting can be simulated by amplitude and phase changes in the gear mesh stiffness model. The gear mesh model developed can easily be incorporated into the global transmission system for dynamic predictions.

3) Using the time averaging technique, frequency spectrum analysis, and the Wigner-Ville distribution, a signature analysis scheme can be developed to examine and characterize the vibration signal of the gear system.

4) A parametric study of the effects on the vibration signal due to various changes in the gear mesh stiffness model, simulating various degrees of pitting and wear damage, could provide a comprehensive database for gear fault detection and damage estimation research.

CHAPTER 6

QUANTIFICATION OF GEAR TOOTH DAMAGE BY OPTIMAL TRACKING OF VIBRATION SIGNATURE

6.1. Objective

In the last two decades, with demands for higher operating speeds and greater load capacity, premature failures in high-performance turbomachinery have often resulted in enormous financial losses and, at times, catastrophic consequences. In aeronautical applications, where both weight and efficiency are pushed to their design limits, the prevention and management of premature equipment failures is a vital part of the maintenance program. Current onboard condition-monitoring systems for gas turbine engines often fail to provide sufficient time between warning and failure for safety procedures to be implemented. On the other hand, inaccurate interpretation of operating conditions may result in false alarms and unnecessary repairs and downtime. The early detection of incipient failure in a mechanical system is of great practical importance as it permits scheduled inspections without costly shutdowns and indicates the urgency and locations for repair before a system incurs catastrophic failure.

Some success has been achieved in identifying damage in a gear transmission system by using the Wigner-Ville distribution (WVD) technique as described in the previous chapters [4,7,9,15]. The approach is to use statistical pattern recognition to match the WVD signature patterns of damaged gears with standard patterns stored in a data base. Although the WVD technique is useful for determining the type and location of the damage, it is not much help in quantifying the level of damage. Damage quantification would logically be the next step in

failure prediction; however, no explicit attempts at damage quantification have previously appeared in the literature.

This chapter presents a new technique for processing vibration data to quantify the level of damage in a gear transmission system. The technique consists of a nonlinear numerical optimization in the form of an "optimal tracking" problem [56,57]. The optimization uses a dynamic model of the gear mesh and forms an estimate of the time-varying mesh stiffness that best corresponds to the given set of vibration data. The utility of the technique relies on the relationship between the wear or damage of a gear tooth and the change in stiffness of the mesh during a given tooth pass cycle. An analysis of this relationship demonstrates that the perturbation of the stiffness function from the nominal profile can be used to quantify the level of damage.

The optimal tracking technique for estimating the perturbation of the mesh stiffness was tested in two settings. First, it was tested on a set of fictitious data generated by computer simulation of a one-degree-of-freedom mechanical system with time-varying stiffness. The solution of the optimal tracking problem matched very closely the actual stiffness profile used in the model generating the data. Then, the technique was tested on a set of experimental data from a gear test rig, but still assuming the one-degree-of-freedom model. Despite the simplicity of the model the stiffness profile obtained was shown to be useful in correlating to the level of damage of the gear transmission system.

This chapter is organized as follows: Section 2 presents the system model and formulates the optimal tracking problem. Section 3 outlines the numerical solution procedure for the

nonlinear optimization. Section 4 presents and interprets the results of the optimization and discusses the next steps to be taken in developing a comprehensive failure-prediction procedure.

6.2. OPTIMAL TRACKING PROBLEM

6.2.1 System Model

The system considered in this study consisted of a small pinion in mesh with a larger gear. A simple model of this system has the two gear masses connected by a spring and a damper. The larger gear is much heavier than the pinion; hence, it is assumed to be rigid, so that all relative motion between the two is attributed to the motion of the pinion. Then, the equation of motion of the pinion takes the form

$$m\ddot{x} + c\dot{x} + k(t)x = 0 \tag{6.1}$$

where m is the mass of the pinion and $k(t)$ and c are the stiffness and damping of the mesh. The mesh stiffness is not constant but is nominally a periodic function of the gear angle, with each period corresponding to one tooth pass. The high points on the periodic stiffness function correspond to gear angles where two pairs of gear teeth are in contact, and the low points correspond to angles where only one pair is in contact.

It has been found in experiments on gearbox vibrations[7-9] that the gear mesh stiffness changes with the wear, pitting, or fracture of the gear teeth. Such changes in the gear mesh stiffness inevitably lead to changes in the vibration signatures of the mechanical system. The objective of the optimal tracking procedure developed in this study is to reconstruct the true stiffness profile for a damaged gear tooth from the measured vibration. That is, the objective is to

determine the function $k(t)$ that would result in the measured vibration according to the system model (6.1).

The true stiffness profile can be expressed as the sum of a constant (time averaged) component, a nominal periodic component, and a perturbation resulting from gear wear or damage. Accordingly, the system model (6.1) is written as

$$m\ddot{x} + c\dot{x} + [\, k_{ave} - k_{periodic} - k_{peturb}(t) \,] x = 0 \quad , \tag{6.2}$$

or

$$\ddot{x} + \frac{c}{m}\dot{x} + \Omega^2 x = u(t)x \, , \tag{6.3}$$

where $\Omega^2 = k_{ave}/m$ and $u(t)$ represents the total time-varying component of the stiffness divided by the pinion mass. By defining the variables

$$x_1 = \dot{x}, \; x_2 = x, \tag{6.4}$$

the system model can be written in the state-variable form

$$\begin{aligned} \dot{x}_1 &= -\frac{c}{m}x_1 - \Omega^2 x_2 + u(t)x_2 \\ \dot{x}_2 &= x_1 \end{aligned} \tag{6.5}$$

with the given initial conditions

$$x_1(t_0) = \dot{x}_0, \; x_2(t_0) = x_0 . \tag{6.6}$$

6.2.2 Optimization Problem

Suppose that a data set corresponding to the vibration of the pinion is collected over the interval $[t_0,t_f]$. Let the function describing the data set be denoted as $\tilde{x}_2(t)$, since it corresponds to

the modeled variable x_2 (t). The objective is to determine a reasonable time-varying stiffness component $u(t)$ for which the model output x_2 (t) approximates the measured data $\tilde{x}_2(t)$.

A diagram depicting the functional objective is shown in Fig. 60. In the figure $u(t)$ is depicted as an input to be chosen so that the error $e(t)$ will be small for all time. Note that this problem has the form of a tracking problem, where the control input of a system is designed so that the system output follows a prescribed reference function. Such a problem may be approached by using the standard techniques of optimal control theory[56,57]. In particular, the "design" of a suitable function $u(t)$ may be achieved by minimizing the quadratic cost functional

$$J = \tfrac{1}{2}\beta_f\left(x_2(t_f) - \tilde{x}_2(t_f)\right)^2 + \tfrac{1}{2}\int_{t_0}^{t_f}\left\{\beta_1\left(x_2(t) - \tilde{x}_2(t)\right)^2 + \beta_2 u^2(t)\right\}dt, \qquad (6.7)$$

where β_1, β_2, and β_f are cost-function weighting coefficients. This form of the cost functional penalizes the energy in the error between the modeled output and the measured data. It also penalizes the use of too large a stiffness perturbation function in order to avoid singularity in the solution.

In the optimal tracking problem the system dynamic equations (6.5) are treated as equality constraints imposed in the optimization of the cost (6.7). As such, they are appended to the cost function by using time-varying Lagrange multipliers $\lambda_1(t)$ and $\lambda_2(t)$. These Lagrange multipliers are themselves governed by differential equations called the costate equations. The costate equations together with the state equations of the system model form a two-point boundary value problem (TPBVP)[56,57]. The TPBVP equations are

(State equations)

$$\begin{aligned} \dot{x}_1 &= -\frac{c}{m}x_1 - \Omega^2 x_2 + u(t)x_2 \\ \dot{x}_2 &= x_1 \end{aligned} \qquad (6.8)$$

(Costate equations)

$$\dot{\lambda}_1 = -\lambda_2 + \frac{c}{m}\lambda_1$$
$$\dot{\lambda}_2 = \Omega^2\lambda_1 - u(t)\lambda_1 - \beta_1(x_2(t) - \tilde{x}_2(t)) \quad \textbf{(6.9)}$$

(Stationarity condition) $$0 = \lambda_1 x_2 + \beta_2 u(t) \quad (6.10)$$

(Endpoint conditions) $$x_1(t_0) = \dot{x}_0\ ,\ x_2(t_0) = x_0 \quad \textbf{(6.11)}$$

$$\lambda_1(t_f) = 0\ ,\ \lambda_2(t_f) = \beta_f(x_2(t_f) - \tilde{x}_2(t_f))\ . \quad \textbf{(6.12)}$$

The TPBVP (6.8–6.12) represents a set of necessary conditions for $u(t)$ to be the solution of the optimal tracking problem. The TPBVP consists of a set of four coupled differential equations (6.8–6.9), together with an algebraic relation (6.10), and some endpoint conditions (6.11–6.12) at both t_0 and t_f. Notice that the TPBVP is nonlinear: the unknown function $u(t)$ multiplies other independent variables in the differential equations.

6.3. NUMERICAL SOLUTION PROCEDURE

The nonlinear TPBVP (6.8–6.12) is solved by an iterative procedure. A complete and general derivation of the procedure is given in Sage[57] and Dyer and McReynolds [55]. Some of the salient points are outlined below for convenience.

6.3.1 Successive Sweep Method

Solving the nonlinear TPBVP requires an iterative method. Although several approaches are possible, a common and useful one is to begin with an initial guess $u^0(t)$ and use it to integrate the nonlinear state equations (6.8) forward in time starting from the initial conditions (6.11) to determine the nominal state functions $x_1^0(t)$ and $x_2^0(t)$. Then, starting from the final conditions (6.12), integrate the nonlinear costate equations backward in time to determine the

nominal costate functions $\lambda_1^0(t)$ and $\lambda_2^0(t)$. The nominal functions $u^0(t)$, $x_1^0(t)$, $x_2^0(t)$, $\lambda_1^0(t)$, and $\lambda_2^0(t)$ then satisfy all the TPBVP equations except the stationarity condition (6.10).

The nominal state, costate, and input functions must be iteratively updated, so that they will eventually satisfy all the nonlinear TPBVP equations, including the stationarity condition. Each update is accomplished by solving a linearized version of the TPBVP. A standard method for doing this is known as the sweep method, whereby a linear relationship between the state and costate functions is assumed. Then, the linear TPBVP degenerates into a set of ordinary differential equations with endpoint conditions at the final time only. These are solved by a straightforward numerical integration. In the case of the optimal tracker these ordinary differential equations take the form of the coupled Riccati equations

$$\begin{aligned}
\dot{p}_{11} &= 2\,(\frac{c}{m}\,p_{11} - p_{12}\,) + p_{11}^2\frac{x_2^2}{\beta_2} \\
\dot{p}_{12} &= \frac{c}{m}p_{12} - p_{22} - p_{11}(\,-\Omega^2 + u(t) - \frac{1}{\beta_2}\lambda_1 x_2\,) + p_{11}p_{12}\frac{x_2^2}{\beta_2} \\
\dot{p}_{22} &= -2p_{12}\,(\,-\Omega^2 + u(t) - \frac{1}{\beta_2}\lambda_1 x_2\,) + p_{12}^2\frac{x_2^2}{\beta_2} - (\,\beta_1 + \frac{\lambda_1^2}{\beta_2}\,)
\end{aligned} \tag{6.13}$$

with endpoint conditions

$$p_{11}(t_f) = p_{12}(t_f) = 0, \quad p_{22}(t_f) = \beta_f \tag{6.14}$$

together with the auxiliary linear equations

$$\begin{aligned}
\dot{h}_1 &= h_1\,(\,\frac{c}{m} + p_{11}\frac{x_2^2}{\beta_2}\,) - h_2 - p_{11}\varepsilon\frac{x_2}{\beta_2}(\lambda_1 x_2 + \beta_2 u(t)) \\
\dot{h}_2 &= -h_1\,(\,-\Omega^2 + u(t) - \frac{1}{\beta_2}\lambda_1 x_2 - p_{12}\frac{x_2^2}{\beta_2}\,) - \frac{\varepsilon}{\beta_2}(\lambda_1 x_2 + \beta_2 u(t))(p_{12}x_2 + \lambda_1)
\end{aligned} \tag{6.15}$$

with the endpoint conditions

$$h_1(t_f) = h_2(t_f) = 0. \tag{6.16}$$

Note that the x and λ variables in the differential equations (6.13) and (6.15) represent the given nominal functions. (The zero superscripts have been omitted for convenience.) They are simply treated as time-varying coefficients in the numerical integration of the differential equations. The solutions of equations (6.13) and (6.15) are then used to compute the corrections to the nominal state, costate, and input functions. This computation requires yet another numerical integration, this time of the linearized state equations

$$\begin{aligned}\Delta\dot{x}_1 &= -\Delta x_1 \left(\frac{c}{m} + p_{11}\frac{x_2^2}{\beta_2} \right) + \Delta x_2 \left(-\Omega^2 + u(t) - \frac{1}{\beta_2}\lambda_1 x_2 - p_{12}\frac{x_2^2}{\beta_2} \right) - \\ &\quad - h_1\frac{x_2^2}{\beta_2} + \varepsilon\frac{x_2}{\beta_2}(\lambda_1 x_2 + \beta_2 u(t)) \qquad (6.17) \\ \Delta\dot{x}_2 &= \Delta x_1\end{aligned}$$

with the zero initial conditions

$$\Delta x_1(t_0) = \Delta x_2(t_0) = 0. \tag{6.18}$$

Finally, the update of the nominal control is computed as

$$\Delta u = \frac{\varepsilon}{\beta_2}(\lambda_1 x_2 + \beta_2 u(t)) - \frac{1}{\beta_2}(\lambda_1 \Delta x_2 + x_2(p_{11}\Delta x_1 + p_{12}\Delta x_2 + h_1)), \tag{6.19}$$

where ε is the step size, and the new nominal control is given by

$$u^{i+1}(t) = u^i(t) + \Delta u^i(t). \tag{6.20}$$

(The superscripts i and i+1 denoting the iteration number have been reinserted in equation (6.20).) The procedure is repeated until the nominal functions converge to a solution.

The real scalar $\varepsilon \in [0,1]$ in equations (6.15), (6.17), and (6.19) is used as a "step size" parameter. Using a smaller value of ε tends to decrease the magnitudes of the corrections, thereby improving the stability of the iterative procedure but slowing the convergence to the solution. Using a larger value of ε has the opposite effects.

6.3.2 Numerical Details

The choices of the cost-function weighting coefficients β_1, β_2, and β_f are important for effective numerical optimization. The parameter β_1 defines the penalty on the difference between the calculated and reference vibration signals. Since the goal is to minimize the difference between the calculated and tracked vibration signals, a large weighting coefficient β_1 should be chosen. The parameter β_2 defines the penalty on the function $u(t)$. Generically speaking, β_2 should impose a lighter penalty on $u(t)$ than β_1 imposes on the tracking error. Note also that the choice of units has an effect on the appropriate relative sizes of β_1 and β_2. In the examples studied the numerical values of $u(t)$ are considerably larger in magnitude than those of a reasonable vibration-signal error; therefore, even if equal weighting between error and control were desired, β_2 should be chosen to be considerably smaller than β_1. An inappropriately large choice of the parameter β_2 would make the cost function almost unchanged from one iteration to the next. Thus, a small constant value was chosen for the parameter β_2. The parameter β_f defines

the penalty for the error at the final time. If β_f is too small, a large vibration error at the final time will result.

By following these general guidelines the optimization algorithm described in the previous section was realized in a computer program. The equations were integrated with a seventh-order Runge-Kutta-Fehlberg method. A summary of the programming steps is given below :

0. Set $i = 0$ and take the initial guess $u^0(t)$ for the function $u(t)$ to be zero.
1. Using the function $u^i(t)$ from the previous step, integrate the state equations (6.8) forward in time. Calculate the resulting cost function J^i.
2. Integrate the costate equations (6.9) backward in time.
3. Use the computed state and costate variables as time-varying coefficients in the integration of the Riccati equations (6.13) along with the auxiliary equations (6.15) backward in time.
4. Integrate the linearized state equations (6.17) forward in time. Using the linearized stationarity condition (6.19), calculate the correction $\Delta u^i(t)$ to the nominal function $u^i(t)$ and hence the updated function $u^{i+1}(t)$. Also, calculate the new cost function J^{i+1}.
5. Make decisions about the continuation of the optimization procedure and the choice of the parameters:

   a. If the difference between the calculated and tracked vibration signals is small, the optimization procedure is finished.

   b. If the difference $J^{i+1} - J^i < 0$ is large enough, repeat from step 1.

   c. If the difference $J^{i+1} - J^i < 0$ is too small, increase the weighting β_1 and repeat from step 1.

d. If $J^{i+1} > J^i$, repeat from step 1 using a smaller value of the step size ε. If this is not successful, increase the error weighting β_1 and repeat from step 1.

Some comments should be made on step 5 of the numerical procedure. It was observed that for given values of weighting coefficients and the step-size parameter the optimization procedure converges to some value of the cost function. In this case the difference between the values of the cost functions $J^{i+1} - J^i$ becomes negligible after some iterations. This means that the cost associated with the control $u(t)$ is becoming dominant. Therefore, it makes sense to start a new iteration with an increased weight β_1 (i.e., imposing a higher penalty on the vibration error).

6.4. DISCUSSION OF RESULTS

To demonstrate the optimal tracking procedure described above, two numerical cases were used in this study. The first case was a numerical experiment in which the tracker was applied to a set of vibration signals generated numerically, assuming a given gear mesh stiffness profile. The mesh stiffness profile evaluated by the optimal tracker was compared with the original stiffness used in generating the vibration signal. Figure 61(a) shows the comparison between the vibration signal generated by a sinusoidal stiffness and that simulated by the optimal tracker. As shown in the figure the two vibration signals were very similar; the small difference between the two signals is given in Fig. 61(b). Figure 62(a) depicts the original gear mesh stiffness used and the stiffness evaluated by using the optimal tracker; the difference between the two stiffnesses is given in Fig. 62(b). The excellent agreement between the two stiffnesses in this

numerical experiment has confirmed the applicability of the optimal tracking procedure in evaluating system stiffness changes from system vibration signals. However, this close resemblance between the generated and simulated signals was partly due to the original time signals being smooth, continuous, and harmonic without any substantial change in magnitude and phase over the gear revolution. To demonstrate the generality and limitation of the developed procedure, a set of experimental data taken from a test rig was used in the next case.

The second case was based on the experimental data obtained from the spiral bevel gear test rig shown in Fig. 63. The primary purpose of this rig is to study the effects of gear tooth design, gear materials, and lubrication types on the fatigue strength of aircraft-quality gears [45]. Because spiral bevel gears are used extensively in helicopter transmissions to transfer power between nonparallel intersecting shafts, using this fatigue rig for diagnostic studies is extremely practical. Vibration data from an accelerometer mounted on the pinion shaft bearing housing were captured by using a personal computer with an analog-to-digital conversion board and an anti-aliasing filter. The 12-tooth test pinion and the 36-tooth gear have the following characteristics: 0.5141 in pitch, 35° spiral angle, 1-in. face width, 90° shaft angle, and 22.5° pressure angle. The pinion transmits 720 hp at a nominal speed of 14 400 rpm. The test rig was started and stopped several times for gear damage inspection. The test was ended at 17.72 operational hours when a broken portion of a tooth was found visually during one of the shutdowns.

Figure 64(a) depicts the gear tooth after 6.5 hr of operation. Note that there is heavy surface pitting on one gear tooth with minor pitting on the next tooth. Figure 64(b) shows the time domain averaging, the frequency spectrum, and the joint time-frequency analysis using the

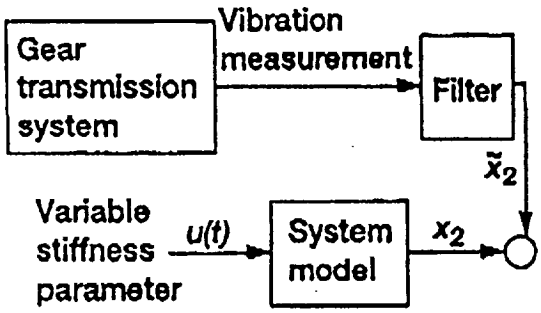


Figure 60 - Block diagram of tracking problem

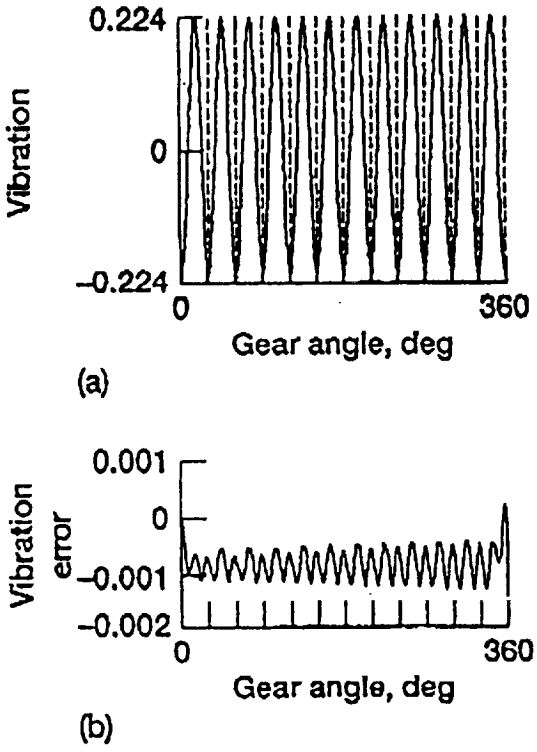


Figure 61 - Original and tracker simulated vibration signals for numerical experiment and error between them (a) Signals (b) error.

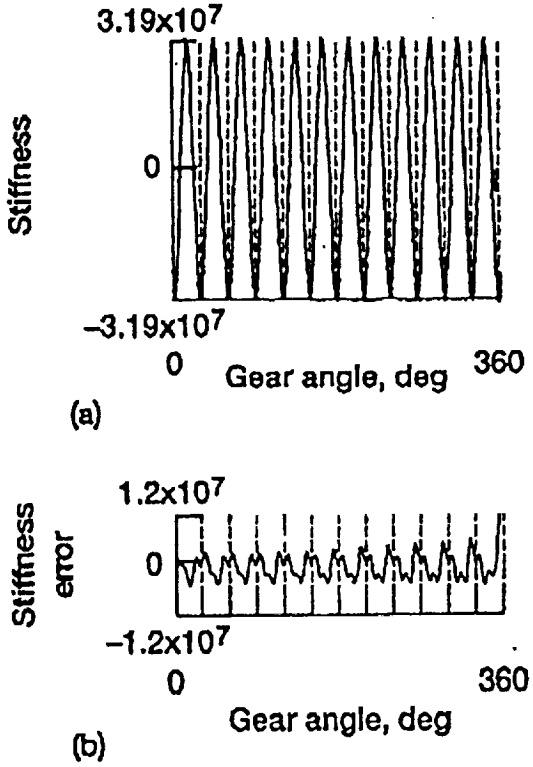


Figure 62 - Original and tracker evaluated mesh stiffness for numerical experiment and error between them, (a) stiffness, (b) error.

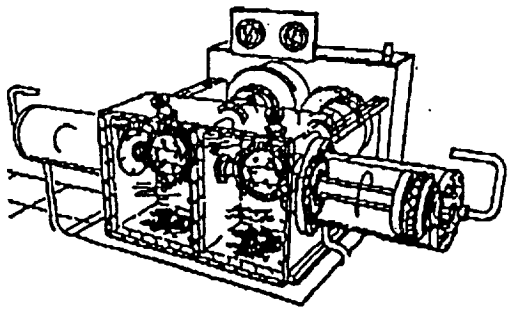


Figure 63 - Experimental spiral bevel gear test rig.

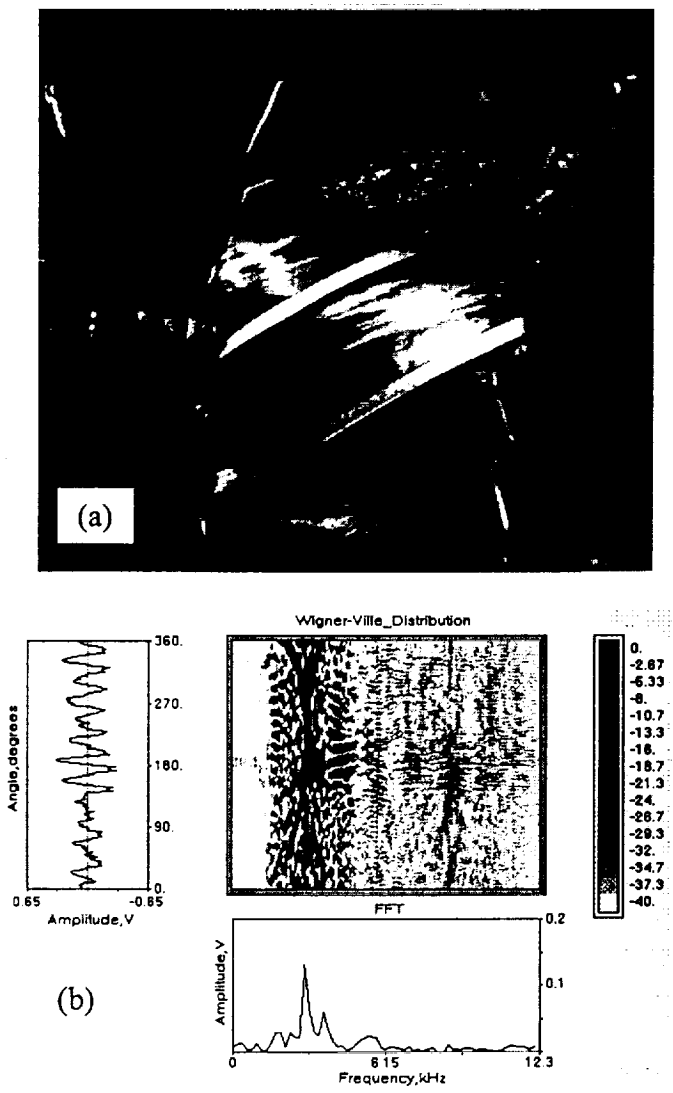


Figure 64. (a) Photograph of damaged gear at 6.55 hr. (b) Gear vibration signal at 6.55 hr, Fast Fourier Transform and Wigner-Ville Distribution.

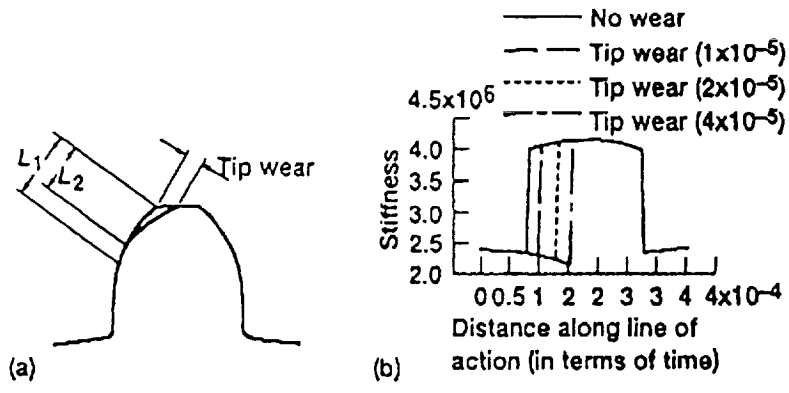


Figure 65. Simulated gear tooth wear and corresponding gear stiffness (a) Simulated gear tooth wear L2:L1 = percent engagement profile wear. (b) Gear stiffness calculated from simulated tooth wear.

Wigner-Ville distribution (WVD) [4,7-9,15] of the accelerometer signal at 6.5 hr[9]. Note that in Fig. 64(b) the time signal indicates a large vibratory signal during the engagement of the sixth and seventh teeth (damaged teeth), but the frequency spectrum, because of its averaging characteristics, shows very little change from the original signal [9]. The WVD begins to show a pattern of shifting of the major frequency component (at a mesh frequency of 2880 Hz) around the meshing of the sixth and seventh teeth. The WVD pattern in this case is very similar to those resulting from a short-term amplitude and phase change of a vibration signal. Although it has been established by the authors in some previous publications [7-9] that such damage in the gear can be identified by the WVD pattern recognition process, the level of the damage has not been addressed. A recent study by the authors has shown that wear and surface pitting of the gear tooth usually will result in a phase shift in the stiffness profile, without any significant change in the stiffness magnitude. Figure 65 shows the stiffness change in a gear mesh evaluated [10] from gear tooth surface profile variations. Note in Fig. 65(b) that increasing surface profile variation increases the phase shift of the gear stiffness without changing the magnitude of the stiffness.

Incorporating this constant gear mesh stiffness as an additional constraint, the optimal tracking procedure was applied to the experimental vibration signal (obtained from the bevel gear test rig at 6.5 hr as shown in Fig. 64) to evaluate the corresponding gear mesh stiffness. To better evaluate the gear mesh stiffness, the time signal was filtered at a mesh frequency of 2880 Hz. Figure 66(a) shows the comparison between the unfiltered experimental signal and the optimal tracker simulation, and Fig. 66(b) shows the comparison between the filtered experimental signal and the tracker-simulated signal. Note that because of the substantial change of magnitude and phase of the time signal during the data acquisition period (one revolution of the gear), the

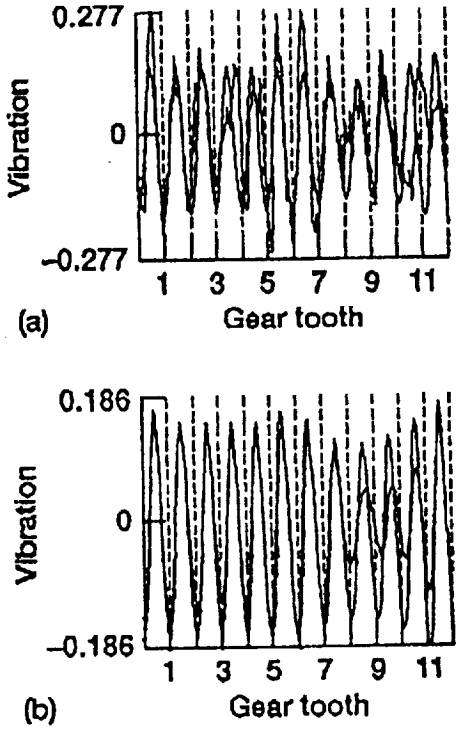


Figure 66 - Filtered and unfiltered experimental and tracker simulated vibration signals for spiral bevel gear at 6.5 hr. (a) Unfiltered. (b) Filtered.

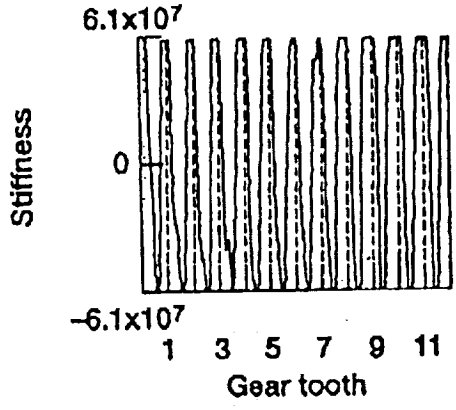


Figure 67 - Tracker evaluated mesh stiffness for spiral bevel gear at 6.5 hr.

accuracy in the simulated vibration is not as good as that in the numerical experiment (Fig. 61(a)). Figure 67 depicts the gear mesh stiffness evaluated by using the optimal tracker. Note that in the evaluated stiffness considerable phase shifts at several gear teeth resulted in the large variation in magnitude and phase of the vibration signal. At the location where pitting occurred (teeth 6 and 7) the phase shift of the stiffness was more pronounced. By using the results from the evaluated mesh stiffness and the correlation of stiffness change with gear wear shown in Fig. 65(b), the gear damage can be estimated.

6.5. CONCLUSIONS

This chapter presents a unified approach to identifying and quantifying damage in a gear transmission system. The conclusions from this study are as follows:

1. The application of the joint time-frequency technique called the Wigner-Ville distribution provides the ability to identify the types and locations of the gear damage.

2. The optimal tracker developed in this chapter provides a very reasonable estimate of the stiffness change due to damage, which can be related to the level of gear damage.

3. For vibratory signals with large changes in magnitude and phase angle the accuracy of the simulated signal from the optimal tracker may decrease.

4. For a more accurate evaluation of system mesh stiffness an optimal tracker for the complete dynamic model of the system is needed.

CHAPTER 7

GENERAL CONCLUSIONS

During this study, a numerical procedure has been developed to simulate the dynamics of gear transmission systems with the effects of gear tooth damage due to wear and pitting. The numerical procedure developed consists of (a) a numerical model to simulate the gear mesh stiffness change resulting from gear tooth damage due to surface pitting and wear, (b) a modal synthesis methodology to simulate the dynamics of gear transmission systems, (c) a joint time-frequency analysis using the Wigner-Villle distribution (WVD) method to provide a comprehensive representation of the vibration signature for fault detection, and (d) the use of an optimal tracker to quantify the damage in the gear tooth. The developed numerical procedure was verified by comparison with vibration data from a damaged gear obtained at the NASA Lewis Research Center. The specific accomplishements in this project can be summarized as follows:

1. A method has been developed to simulate the effects of tooth surface wear in a gear transmission system. The tooth surface wear level can be controlled by adjustment of both amplitude and length in the tooth profile modification.
2. Using the developed gear tooth damage model, tooth damage due to wear and pitting can be simulated by amplitude and phase changes in the gear mesh stiffness model. The gear mesh model developed can easily be incorporated into the global transmission system for dynamic predictions.
3. A numerical procedure has been developed to simulate the vibration in a gear transmission system with effects of gear tooth damage due to wear and pitting. The modal synthesis methodology was used for simulating the dynamics of gear transmission systems. The gear

mesh model developed to simulate the gear tooth damage due to wear and pitting are incorporated numerically into the global system for dynamics predictions.

4. The Wigner-Villle distribution (WVD) method used for the joint time-frequency analysis provides a comprehensive representation of the vibration signal. It was successfully used to verify the numerical model developed.
5. Based on the results of the application of the various time and frequency analysis techniques, it can be concluded that the WVD provides vital information concerning both the severity and the location of the pitting process in the gear system. The fracture of the gear tooth and its exact location can be pinpointed using the WVD technique. However a machine vibration signature database is required to interpret the resulting WVD.
6. A parametric study of the effects on the vibration signal due to various degrees of pitting and wear damage, has provided a comprehensive database for gear fault detection and damage estimation research.
7. A parametric study of the effects on the vibration signal due to various changes in the gear mesh stiffness model, simulating various degrees of pitting and wear damage, could provide a comprehensive database for gear fault detection and damage estimation research.
8. The optimal tracker developed in this chapter provides a very reasonable estimate of the stiffness change due to damage, which can be related to the level of gear damage. For vibratory signals with large changes in magnitude and phase angle the accuracy of the simulated signal from the optimal tracker may decrease. For a more accurate evaluation of system mesh stiffness an optimal tracker for the complete dynamic model of the system is needed.

REFERENCES

1. Coy, J.J., Townsend, D.P., Zaretsky, E.V., "Dynamic Capacity and Surface Fatigue Life for Spur and Helical Gears", ASME Journal of Lubrication Technology, Vol. 98, No.2, April 1976, pp. 267-276

2. Savage, M., Radil, K.C., Lewicki, D.G., and Coy, J.J., "Computerized Life and Reliability Modeling for Turbo-Prop Transmissions", AIAA Jounal of Propulsion and Power, Vol.5, No.5, 1989

3. Lundberg, G., Palmgren, A., "Dynamic Capacity of Roller Bearings", ACTA Polytechnica, Mechanical Engineering Series, Vol.2, No.4, 1952

4. Shives, T.R., Mertaugh, L.J.(Editor), "Detection, Diagnosis and Prognosis of Rotating Machinery to Improve Reliability, maintainability, and Rediness through the Application of New and Innovative Techniques", Proceedings of the 41st MFPG Meeting in Naval Air Test Center, Patuxent River, Maryland, October 28-30, 1986

5. Pusey, H.C.(Editor), "The System Engineering Approach to Mechanical Failure Prevention", Proceedings of the 41st Mechanical Failure Prevention Group, Office of Naval Research, Arlington, Virginia, 1993

6. Pusey, H.C.(Editor), " Advanced Material and Process Technology from Mechanical Failure Prevention", Proceedings of the 42st Mechanical Failure Prevention Group, Office of Naval Research, WAkefield, Massachusetts, April 19-21, 1994

7. Choy, F.K., Braun, M.J., Polyshchuk, V., Zakrajsek, J.J., Handschuh, R.F., Townsend, D.P., "Analytical and Experimental Vibration Analysis of a Damaged Gear System", Presented in the 1994 AGMA Fall Technical Meeting, St. Louis, Missouri, October 24-26, 1994.

8. Choy, F.K., Polyshchuk, V., Zakrajsek, J.J., Handschuh, R.F., Townsend, D.P., "Analysis of the Effects of Surface Pitting and Wear on the Vibrations of a Gear Transmission System', Presented in the 4th International Tribology Conference, Perth, Western Australia, December 5-8, 1994

9. Choy, F.K., Huang, S., Zakrajsek, J.J., Handschuh, R.F., Townsend, D.P., " Vibration Signature Analysis of a Faulted Gear Transmission System", Paper presented in the 30th AIAA Joint Propulsion Conference, Indianapolis, Indiana, June 27-29, 1994

10. Boyd, L.S., Pike, J., "Multi-Mesh Gear Dynamics Program Evaluation and Enhancements", NASA Contractor Report 174747, prepared for Lewis Research Center under Contract NAS3-23928, June 1985

11. Choy, F.K., Tu, Y.K., Savage, M., Townsewnd, D.P., "Vibration Signature Analysis of Multistage Gear Transmission", Journal of the Franklin Institute, Vol. 328, No.2/3, 1991

12. Choy, F.K., Ruan, Y.F., Zakrajsek, J.J., Oswald, F.B., "Modal Simulation of Gearbox Vibration with Experimental Correlation", AIAA Journal of Propulsion and Power, Vol. 9, No.2, March 1993

13. Kahraman, A., Ozguven, H.N., Houser, D.R., Zakrajsek, J.J., "Dynamic Analysis of Geared Rotors by Finite Element", NASA TM-102349, AVSCOM-TM-89-C-006, 1990

14. Boashash, B., Black, P.J., "An Efficient Real Time Implementation of the WingerVille Distribution", IEEE Trans. on Acoustics, Speech, and Signal Procesing, Vol. ASSP35, No.11, November 1987

15. Claasen, T.A.C.M., Mecklenbruker, W.F.G., "The Wigner Distribution A Tool for TimeFrequency Signal Analysis", PartI Philip J. Res. 35, 1980

16. Shin, Y.S., Jeon, J.J., "Pseudo WignerVille TimeFrequency Distribution and its Application to Machinery Condition Monitoring", Journal of Shock and Vibration, Vol.1, Issue 1, 1993/1994, pp. 65

17. Forrester, B.D., "Analysis of Gear Vibration in the TimeFrequency Domain", Proc. of the 44th Meeting of the Mechanical Failure Prevenention Group, Feb. 1990

18. Mcfadden, P.D. and Wang, W.J., "TimeFrequency Domain Analysis of Vibration Signal for Machinery Diagnostics (II) the Weighted WignerVille Distribution", University of Oxford, Report No. OUEL 1891, 1991

19. Richardson, H.H., "Static and Dynamic Load, Stress, and Deflection Sycles in Spur-Gear Systems", Sc.D Thesis, M.I.T. Report, 1958

20. Cornell, R.W., "Compliance and Stress Sensitivity of Spur Gear Teeth", Journal of Mechanical Design, Vol. 103, April 1981

21. Weber, C., "The Deformations of Loaded Gears and the Effect on Their Load-Carrying Capacity", Sponsored Research (Germany), British Dept. of Scientific and Industrial Research, Report No.3, 1949

22. O'Donnell, W.J., "Stress and Deflection of Built-in Beams", ASME Paper No. 62-WA-16, 1974

23. Metusz, J.M., O'Donnell, W.J., Erdloc, R.J., "Local Flexibility Coefficients for the Built-in Ends of Beams and Plates", ASME, Journal of Engineering for Industry, August 1969

24. Mcfadden, P.D., " Examination of a Technique for the early Detection of Failure in Gears by Signal Processing of the Time Domain Average of the Meshing Vibration", Mechanical System and Signal Processing, January 1, 1987, pp. 173-183

25. Choy, F.K., Veillette, R.J., Polyshchuk, V., Braun, M.J., Hendricks, R., "Quantification of Gear-tooth Damage via Optimal Tracking of Vibration Signatures', Paper submitted for publication in the 6th International Symposium on Transprot Phenomena and Dynamics of Rotating Machinery , Honolulu, Hawaii, Fabruary 25-29, 1996

26. Mcfadden, P.D., " Examination of a Technique for the early detection of Falure in Gears by Signal Processing of the Time Domain Average of the Meshing Vibration", Mechanical Systems and Signal Processing, Issue 2, 1987, pp.173

27. Lin, H.H., Townsend, D.P., Oswald, F.B., "Profile Modification to Mimimize Spur Gear Dynamic Loading", NASA TM-89901, 1988

28. Cornell, R.W., Westervelt, W.W., "Dynamic Tooth Loads and Stressing for High Contact Ratio Spur Gears", Jounal of Mechanical Design, Vol.100, January 1978

29. Choy, F.K., Qian, W., "Global Dynamic Modeling of a Transmission Systems", NASA Contractor Report 191117, ARL Contractor Report ARL-CR-11, April 1993

30. Lin, H.H., Wang, J., Oswald, F.B., and Coy, J.J., "Effect of Extended Tooth Contact on the Modeling of Spur Gear Transmissions", Paper presented in the 29th Joint Propulsion Conference and Exhibit , Monterey, California, June 28-30, 1993

31. Choy, F.K., Townsend, D.P., and Oswald, F.B., "Experimental and Analytical Evaluation of Dynamic Load and Vibration of a 2240-kW (3000-hp) Rotorcraft Transmission", Paper presented in the Design Engineering Conference and Show, Chicago, Illinois, March 2-5, 1987

32. Choy, F.K., Ruan, Y.F., Zakrajsek, J.J., Oswald, F.B., and Coy, J.J., "Analytical and Experimental Study of Vibrations in a Gear Transmission", Paper presented in the 27th Joint Propulsion Conference, Sacramento, California, June 24-27, 1991

33. Choy, F.K., Tu, Y.K., Zakrajsek, J.J., and Townsend, D.P., "Dynamics of Multistage Gear Transmission With Effects of Gearbox Vibrations", Paper presented in the CSME Mechanical Engineering Forum 1990, Toronto, Canada, June 3-9, 1990

34. Choy, F.K., Ruan, Y.F., Tu, Y.K., Zakrajsek, J.J., and Townsend, D.P., "Modal Analysis of Multistage Gear Systems Coupled With Gearbox Vibrations", Journal of Mechanical Design, Vol. 114, September 1992

35. Lin, H.H., Huston, R.L., "Dynamic Loading on Parallel Shaft Gears", NASA Contractor Report 179473, Prepared for Lewis Research Center Under Grant NSG-3188, July 1986

36. Baronet, C.N., Tordion, G.V., "Exact Stress Distribution in Standard Gear Teeth and Geometry Factors", Jonrnal of Engineering for Industry, November 1973

37. Winter, H., Hirt, M., "The Measurement of Actual Strans at Gear Teeth, Influence of Fillet Radius on Stres and Tooth Strength", Journal of Engineering for Industry, Februry 1974

38. Chabert, G., Tran, T.D., Mathis, R., "An Evaluation of Stresses and Deflection of Spur Gear Teeth Under Strain", Journal of Engineering for Industry, February, 1974

39. Zakrajsek, J.J., "An Investigation of Gear Mesh Failure Prediction Techniques", NASA Technical Memorandum 102340, November 1989

40. Choy, F.K., Townsend, D.P., Oswald, F.B., "Dynamics Analysis of Multimesh-Gear Helicopter Transmissions", NASA Technical Paper 2789, February 1988

41. Peyrin, F., Frost, R., "A Unified Definitionl for the Discrete-Time, Discrete-Frequency, and Discrete-Time/frequency Wigner Distributions", IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-34, No. 4, August 1986

42. Narayanan, S.B., Prabhu, K.M.M., "New Method of computing Wigner-Villle Distribution", Elecronics Letters, Vol. 25, No. 5, March 2, 1989

43. Claasen, T.A.C.M., Mecklenbrauker, W.F.G., "The Aliasing Problem in Discrete-Time Wigner Distributions", IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-31, No. 5, October 1983

44. Boashash, B. "On the Anti-Aliasing and Computational Properties of the Wigner-Ville Distribution"

45. Zakrajsek, J.J., Handschuh, R.F., and Decker, H.J., "Application of Fault Detection Techniques to Spiral Bevel Gear Fatigue Data"

46. Stewart, R.M., "Some Useful Data Analysis Techniques for Gearbox Diagnostics," Institute of Sound and Vibration Research, Paper MHM/R/10/77, 1977.

47. Decker, H.J., Handschuh, R.F., and Zakrajsek, J.J., "An Enhancement to the NAU Gear Vibration Diagnostic Parameter," NASA TM–106553, presented at the 18th Annual Meeting of the Vibration Institute, Hershey, PA, June 20–23, 1994.

48. Zakrajsek, J.J., Towsend, D.P. and Decker, H.J. "An Analysis of Gear Fault Detection Methods as Applied to Pitting Fatigue Failure Data", NASA TM–105950, presented at the 47th Mechanical Failure Prevention Group Meeting, Virginia Beach, Virginia, April 13–15, 1993.

49. Bouachache, B. and Flandrin, P., "Wigner-Ville Analysis of Time-Varying Signals," Proc. 2CASSP82, Paris, France, May 1982, pp. 1329-1333.

50. Handschuh, R., "Effect of Lubricant Jet Location on Spiral Bevel Gear Operating Temperatures," NASA TM–105656, AUSCOM TR–91–C–033, presented at the 6th International Power Transmission and Gearing Conference, ASME Sept. 13–16, 1992, Phoenix, AZ.

51. Choy, F.K., Ruan, Y.F., Zakrajsek, J.J., Oswald, F.B.,"Modal Simulation of Gearbox Vibration with Experimantal Correlation" AIAA J. of Propulsion and Power, Vol.9, No.2, March 1993.

52. Kahraman, A.; Ozguven, H.N.; Houser, D.R.; and Zakrajsek, J.J.; "Dynamic Analysis of Geared Rotors by Finite Element", NASA TM-102349, AVSCOM-TM-89-C-006, 1990.

53. Stewart, R.M., " Some Useful Data Analysis Techniques for Gearbox Diagnostics," Institute of Sound and Vibration Research, Paper MHM/R/10/77, 1977.

54. Choy, F.K.; Townsend, D.P.; and Oswald, F.B., "Dynamic Analysis of Multimesh-Gear Helicopter Transmissions', NASA TP-2789, 1988.

55. Dyer, P and Mcreynolds, S. R., 1970, "The computation and Theory of Optimal Control", Academic Press.

56. Lewis, F. L., 1986, "optimal Control", Wiley

57. Sage, A. P. , 1968, "Optimal System Control", Printice-Hall.

APPENDIX

LISTING OF COMPUTER PROGRAM FOR DYNAMIC SIMULATION

```
C************ VLADIMIR V. POLYSHCHUK ********************************C
C************ poly@fcalpha.mechanical.uakron.edu ********************C
C******************  JULY 20, 1995      ****************************C
C******************  Gear Analysis Program ***************************C
C* NEW DEVELOPMENT CYCLE AS A TRANSITION TO PC WINDOWS C++ PROGRAM C
C*****************************************C
C*********************************************C
C*********************************************C
C*************** GUANGHUI XU ***********************************C
C*************** JUNE 7, 1996 **********************************C
C*************** MODIFIED TO INCLUDE DAMAGED GEAR CASE **********C
C*****************************************************************C
C*****************************************************************C
C*****************************************************************C
      PROGRAM GAPD
      PARAMETER N50=50
      PARAMETER N20=20
      PARAMETER N10=10

      CHARACTER*30 DATA_FILE(N50), FSTAT(N50)
      CHARACTER*30 FILE_OUT(N50)
      CHARACTER*30 NAME
      CHARACTER*1 A,B

      INTEGER NROT, NGEARS, NOBOX, NOMODE, ICOUNT
      INTEGER NMODE(N50)
      INTEGER NUNIT(N50),NSTAT(N50)
      INTEGER NBEAR(N10), LBEAR(N10,N10)
      INTEGER NCON_BEAR(N10,N10,N10,N10)
      INTEGER LONG
      INTEGER NUM, NSWITCH(N10)
      REAL STIF2(N10,N10,6,6),STIFA(N10,N10,6,6)
      REAL DAMP2(N10,N10,6,6)
      REAL SHAPE(N50,N50,N50,4)
      REAL FREQ(N10,N10,N10)

C*****************************************************************C
      OPEN (9,FILE='menu.dat',STATUS='OLD')

      DO 100 I=1,N10
100   NSWITCH(I)= 0

      DO 10 I=1,N50
      NUNIT(I)=9 + I
10    CONTINUE

      ICOUNT= 0
      READ (9,*)
      READ (9,*) NROT
      READ (9,*)
      READ (9,*) NGEARS
      READ (9,*)
C* ENTER FILE WITH SHAFT DATA *C
      READ (9,15) DATA_FILE(ICOUNT+1)
      FSTAT(ICOUNT+1)='OLD'
      READ (9,*)
C* ENTER FILE WITH GEAR DATA *C
      READ (9,15) DATA_FILE(ICOUNT+2)
      FSTAT(ICOUNT+2)='OLD'
C* ENTER FILE WITH BOX DATA *C
      READ (9,*)
      READ (9,*) NOBOX
      IF ( NOBOX .EQ. 1 ) THEN
      READ (9,*)
      READ (9,*)
      ICOUNT= ICOUNT - 1
```

```
      ELSE
      READ (9,*)
      READ (9,15) DATA_FILE(ICOUNT+3)
      FSTAT(ICOUNT+3)='OLD'
      ENDIF
      READ (9,*)
      READ (9,*) NOMODE
      READ (9,*)
      READ (9,15) DATA_FILE(ICOUNT+4)
      READ (9,*)
      READ (9,15) DATA_FILE(ICOUNT+5)

      IF ( NOMODE .EQ. 1 ) THEN
      FSTAT(ICOUNT+4)='NEW'
      FSTAT(ICOUNT+5)='NEW'
      ELSE
      FSTAT(ICOUNT+4)='OLD'
      FSTAT(ICOUNT+5)='OLD'
      ENDIF

      DATA_FILE(ICOUNT+6)='input.out'
      FSTAT(ICOUNT+6)= 'NEW'

      NCOUNT=ICOUNT+6
      DO 30 I=1,NCOUNT
      OPEN(NUNIT(I),FILE= DATA_FILE(I),STATUS= FSTAT(I) )
 30   CONTINUE

      READ (9,*)
      READ (9,15) FILE_OUT(1)

      READ (9,*)
      READ (9,*) NUM
      READ (9,*) ( NSWITCH(I),I= 1,NUM )
      WRITE(NUNIT(NCOUNT),*) ( NSWITCH(I),I= 1,NUM )

C*************************************************************C
C*******  END OF MENU INPUT ***************************C
C*************************************************************C
      DO 32 I=2,LEN(FILE_OUT(1))
      IF( FILE_OUT(1)(I-1:I) .EQ. ' ' ) THEN
      LONG= I-2
      GOTO 33
      ENDIF
 32   CONTINUE
 33   CONTINUE

      DO 35 I=1,NGEARS
      IF( I .LT. 10 ) THEN
      A= CHAR(48+I)
      NAME= FILE_OUT(1)
      OPEN(NUNIT(NCOUNT+I),FILE= NAME(:LONG)//'.'//A,
     &                                        STATUS='NEW')
      ELSE
      K= INT(I/10)
      J= MOD(I,10)
      A= CHAR(48+K)
      B= CHAR(48+J)
      NAME= FILE_OUT(1)
      OPEN(NUNIT(NCOUNT+I),FILE= NAME(:LONG)//'.'//A//B,
     &                                        STATUS='NEW')
      ENDIF
 35   CONTINUE

 15   FORMAT(1X, 25A )
```

```
C*********************************************************************C
      CALL LAT(NUNIT,NCOUNT,NROT,NSTAT,NBEAR,LBEAR,STIF2,
     & DAMP2,STIFA,FREQ,SHAPE,NMODE,NCON_BEAR, NSWITCH)
C*********************************************************************C
      IF( NSWITCH(2) .EQ. 1 ) STOP

      CALL MOTION(NUNIT,NCOUNT,NROT,NSTAT,NBEAR,LBEAR,STIF2,
     & DAMP2,STIFA,FREQ,SHAPE,NMODE,NCON_BEAR, NSWITCH)
C*********************************************************************C
      END
C*********************************************************************C
      SUBROUTINE MOTION(NUNIT,NCOUNT,NROT,NSTAT,NBEAR,LBEAR,STIF2,
     & DAMP2,STIFA,FREQ,SHAPE,NMODE,NCON_BEAR, NSWITCH)

      PARAMETER NONE= 1
      PARAMETER UNBAL= 2
      PARAMETER BEAR= 3
      PARAMETER GEAR= 4
      PARAMETER EXTERN= 5

      PARAMETER ROTOR= 1
      PARAMETER CASING= 2
      PARAMETER GROUND= 3

      PARAMETER N50=50
      PARAMETER NROT_SIZE=10
      PARAMETER N10=10
      PARAMETER N20=20
      PARAMETER N30=30
      PARAMETER N120=120
      PARAMETER N500=500
      PARAMETER N5=5
      PARAMETER N140=140
      PARAMETER PI= 3.14

C* OLD VARIABLES *C
      INTEGER NROT
      REAL SHAPE(N50,N50,N50,4)
      REAL FREQ(N10,N10,N10)

      REAL FTEMP(N10,N10,N10)

      INTEGER NMODE(N50)
      INTEGER NUNIT(N50), NSTAT(N50)
      INTEGER NCOUNT

      INTEGER NBEAR(N10), LBEAR(N10,N10)
      INTEGER NGEAR(N10), LGEAR(N10,N10)
      INTEGER NCON_BEAR(N10,N10,N10,N10)
      INTEGER NCON_GEAR(N10,N10,N10,N10)

      INTEGER  NSWITCH(N10)
      REAL STIF2(N10,N10,6,6), STIFA(N10,N10,6,6)
      REAL DAMP2(N10,N10,6,6)

      INTEGER NFACT,NPOINTS,NCALC,NCYCLE,NCYCLE_TRAN
      INTEGER NUNBAL(N20), ISTAT_UN(N20,N20)
      INTEGER NUNIT_OUT(N20,N20)

      REAL FSAMPLE, TCYCLE(N20)
      REAL UNBX(N20,N20), UNBY(N20,N20), PHIADD(N20,N20)

C* NEW VARIABLES *C
C* GLOBAL VARIABLES *C
      INTEGER NODE_TABLE(N20,5,N50)
      INTEGER CLASS_TABLE(N20,N20)
```

```
      INTEGER NODE_CON(N50,N50)
      INTEGER CLASS_NODE(N50)
      INTEGER CLASS_NAME(N50)
      INTEGER NODE_LOC(N50)

      REAL SHAPE_NODE(N50,N20,6)
      REAL STIF_BEAR(N20,6,6), DAMP_BEAR(N20,6,6)
      REAL STIF_AV(N20,6,6)
      REAL XN(N20,N20,6), VN(N20,N20,6), AN(N20,N20,6)
      REAL XNP(N20,N20,6), VNP(N20,N20,6), ANP(N20,N20,6)
      REAL FBEAR(N20,N20,6)
      REAL FGEAR(N20,N20,6)
      REAL FUNB(N20,N20,6)
      REAL TORQUE(N20), FEXT(N20,N20,6)

      REAL AMPL_PHASE(N50,3)
      REAL ANGULA(N20,2)
      REAL TIME, DELTAT

      INTEGER YES_NO, NL
      INTEGER ITYPE, NTYPE, NTYPE_GLOBAL, NTYPE_STIF
      INTEGER  NPOINT(N20)
      INTEGER ITYPE_DAM(N50,N50)

      REAL AGEAR(N20,N20,N10), AGSTIF(N20,N120)
      REAL STIF_GEAR(N20,N120)
      REAL TMR(N20,3,3)
      REAL ACUR(N20)
      REAL GEAR_STIF

C      AMPL= AMPL_PHASE(INODE,1)
       PHASE= AMPL_PHASE(INODE,2)
       PHIADD= AMPL_PHASE(INODE,3)

C******** INITIALIZATION NCON_GEAR TABLE ******************************C
      DO 5 IROT=1,NROT
      DO 5 IGEAR=1,NGEAR(IROT)
      DO 5 JROT=1,NROT
      DO 5 JGEAR=1,NGEAR(JROT)
      NCON_GEAR(IROT,IGEAR,JROT,JGEAR)=0
 5    CONTINUE

C*********************************************************************C
C************ INPUT GEAR DATA OF THE ROTOR-BEARING SYSTEM *************C
C*********************************************************************C
      INODE= 1
      JCHECK=0
      READ (NUNIT(2),*)
      READ (NUNIT(2),*)
      READ (NUNIT(2),*) NCYCLE, NCYCLE_TRAN, NPOINTS, NFACT
      READ (NUNIT(2),*)
      READ (NUNIT(2),*) NROT
 10   READ (NUNIT(2),*)
      READ (NUNIT(2),*) IROT
      READ (NUNIT(2),*)
      READ (NUNIT(2),*) ANGULA(IROT,1), ANGULA(IROT,2)
      READ (NUNIT(2),*)

      READ (NUNIT(2),*) NUNBAL(IROT)
      DO 500 I=1,NUNBAL(IROT)
      READ (NUNIT(2),*) ISTAT_UN(IROT,I),
     &     UNBX(IROT,I), UNBY(IROT,I), PHIADD(IROT,I)

      WRITE (NUNIT(NCOUNT),*) 'IROT ',IROT,' NUNBAL ',NUNBAL(IROT),
     &                                     ' UNBX ',' UNBY'
      WRITE (NUNIT(NCOUNT),*) UNBX(IROT,I), UNBY(IROT,I)
```

```
500   CONTINUE

      READ (NUNIT(2),*)
      READ (NUNIT(2),*) NODE_LOC(INODE), TORQUE(IROT)
      CLASS_TABLE(IROT,EXTERN)= 1
      NODE_TABLE(IROT,EXTERN,1)= INODE
      CLASS_NODE(INODE)= IROT
      INODE= INODE + 1

      READ (NUNIT(2),*)
      READ (NUNIT(2),*) CLASS_TABLE(IROT,NONE)
      DO 505 INONE= 1, CLASS_TABLE(IROT,NONE)
      READ (NUNIT(2),*) ILOC
      NODE_LOC(INODE)= ILOC
      NODE_TABLE(IROT,NONE,INONE)= INODE
      CLASS_NODE(INODE)=IROT
      INODE= INODE + 1
505   CONTINUE

      JCHECK=JCHECK+1
      IF ( JCHECK .LT. NROT ) GOTO 10

      READ (NUNIT(2),*)
      READ (NUNIT(2),*) YES_NO
      IF( YES_NO .EQ. 0 ) GOTO 510

      WRITE (NUNIT(NCOUNT),*)
      WRITE (NUNIT(NCOUNT),*) ' GEAR DATA INPUT '
      WRITE (NUNIT(NCOUNT),*)

      JCHECK=0
515   READ (NUNIT(2),*)
      READ (NUNIT(2),*) IROT
      READ (NUNIT(2),*)
      READ (NUNIT(2),*) NGEAR(IROT)
      DO 520 IGEAR= 1,NGEAR(IROT)
      READ (NUNIT(2),*) LGEAR(IROT,IGEAR)
      WRITE (NUNIT(NCOUNT),*) LGEAR(IROT,IGEAR)
      READ (NUNIT(2),*) ( AGEAR(IROT,IGEAR,I), I=1,5 )
      WRITE (NUNIT(NCOUNT),*) ( AGEAR(IROT,IGEAR,I), I=1,5 )
520   CONTINUE

      JCHECK=JCHECK+1
      IF ( JCHECK .LT. NROT ) GOTO 515

      WRITE (NUNIT(NCOUNT),*) ' GEAR CONNECTION TABLE '
      WRITE (NUNIT(NCOUNT),*) ' IROT  IGEAR  JROT  JGEAR
     &                    ITYPE_CON '
      READ (NUNIT(2),*)
      READ (NUNIT(2),*) NL
      DO 525 I= 1,NL
      READ (NUNIT(2),*) IROT, IGEAR, JROT, JGEAR, ITYPE
      NCON_GEAR(IROT,IGEAR,JROT,JGEAR)= ITYPE
      NCON_GEAR(JROT,JGEAR,IROT,IGEAR)= NCON_GEAR(IROT,IGEAR,JROT,JGEAR)

      WRITE (NUNIT(NCOUNT),*) IROT, IGEAR, JROT, JGEAR, ITYPE
CC     &                          NCON_GEAR(IROT,IGEAR,JROT,JGEAR)
525   CONTINUE
      READ (NUNIT(2),*)
      READ (NUNIT(2),*)

      JCHECK=0
      READ (NUNIT(2),*) NTYPE_STIF, NTYPE_DAM
530   READ (NUNIT(2),*)
      READ (NUNIT(2),*) ITYPE
      READ (NUNIT(2),*) NL, AGSTIF(ITYPE,1),AGSTIF(ITYPE,2)
```

```
      AGSTIF(ITYPE,3)=0.0
      NPOINT(ITYPE)= NL
      READ (NUNIT(2),*) (AGSTIF(ITYPE,3+I),STIF_GEAR(ITYPE,I),I=1,NL)

      WRITE (NUNIT(NCOUNT),*)
      WRITE (NUNIT(NCOUNT),*) ' DATA FOR GEAR TYPE ',ITYPE
      WRITE (NUNIT(NCOUNT),532)
     &                  (AGSTIF(ITYPE,3+I),STIF_GEAR(ITYPE,I),I=1,NL)

  532 FORMAT(1X, 2F20.4 )

      JCHECK= JCHECK + 1
      IF ( JCHECK .LT. (NTYPE_STIF+NTYPE_DAM) ) GOTO 530

      NTYPE_GLOBAL=1
      NTYPE= NTYPE_STIF + NTYPE_GLOBAL
      READ (NUNIT(2),*)

      DO 535 ITYPE=1,NTYPE
      READ (NUNIT(2),*)
      READ (NUNIT(2),*) ((TMR(ITYPE,IROW,JCOL),JCOL=1,3),IROW=1,3)

      WRITE (NUNIT(NCOUNT),*)
      WRITE (NUNIT(NCOUNT),*) 'DATA FOR COORD. TRANSFORM TYPE ',ITYPE
      WRITE (NUNIT(NCOUNT),533)
     &                        ((TMR(ITYPE,IROW,JCOL),JCOL=1,3),IROW=1,3)
  533 FORMAT( 1X, 3F10.4 )

  535 CONTINUE

C**************************************C
C**** START READING GEAR DAMAGE MODEL *C
C      DO 1999 I=1,NTYPE
C      DO 1999 J=1,N30
C      NDAM(I,J)= I
C1999 CONTINUE

C      READ (NUNIT(2),*)
C      READ (NUNIT(2),*) NTYPE_DAM
C      READ (NUNIT(2),*)
C      WRITE (NUNIT(NCOUNT),*) ' DAMAGE MODEL '
C      DO 2200 I=1,NTYPE_DAM
C      READ (NUNIT(2),*) ITYPE,ITER, NEW_ITYPE
C      NDAM(ITYPE, ITER )= NEW_ITYPE
C      WRITE (NUNIT(NCOUNT),*) ITYPE,ITER,NDAM(ITYPE, ITER )
C2200 CONTINUE
      JCHECK = 0
CC      write(66,*) 'before reading of damage'
 1050 READ (NUNIT(2),*)
      READ (NUNIT(2),*) ITYPE,NTEETH
      DO 1000 I=1,NTEETH
         ITYPE_DAM(ITYPE,I) = ITYPE
 1000 CONTINUE
CC      write(66,*) 'before reading of ndam'
      READ (NUNIT(2),*) NDAM
      WRITE(NUNIT(NCOUNT),*)
      WRITE(NUNIT(NCOUNT),*) 'DAMAGE DATA FOR GEAR TYPE',ITYPE
      WRITE(NUNIT(NCOUNT),*)  'ITEETH  ITYPE_DAM', '   NDAM ', NDAM
      IF ( NDAM .LE. 0) GO TO 1020
      DO 1010 I=1,NDAM
         READ (NUNIT(2),*) ITEETH, ITYPE_IDAM
         ITYPE_DAM(ITYPE,ITEETH) = ITYPE_IDAM
         WRITE(NUNIT(NCOUNT),*) ITEETH, ITYPE_DAM(ITYPE,ITEETH)
 1010 CONTINUE
 1020 JCHECK = JCHECK + 1
      IF (JCHECK .LT. NTYPE_STIF) GO TO 1050
```

```
C*********************************************************************C

C* END OF GEAR INPUT DATA FROM ABASE.DAT *C
 510  CONTINUE

C*********************************************************************C
C*********************************************************************C
      DO 700 IROT=1,NROT
      DO 700 IMODE=1,NMODE(IROT)
      FTEMP(IROT,IMODE,1)= FREQ(IROT,IMODE,1)*0.1047195
      FTEMP(IROT,IMODE,2)= FREQ(IROT,IMODE,1)*0.1047195
      FTEMP(IROT,IMODE,3)= FREQ(IROT,IMODE,1)*0.1047195
      FTEMP(IROT,IMODE,4)= FREQ(IROT,IMODE,1)*0.1047195
      FTEMP(IROT,IMODE,5)= FREQ(IROT,IMODE,3)
      FTEMP(IROT,IMODE,6)= FREQ(IROT,IMODE,4)
 700  CONTINUE

      WRITE(NUNIT(NCOUNT),*)

      DO 710 IROT=1,NROT
      DO 710 IMODE=1,NMODE(IROT)
      DO 710 ICOORD=1,6
      FREQ(IROT,IMODE,ICOORD)= FTEMP(IROT,IMODE,ICOORD)
      WRITE(NUNIT(NCOUNT),*) ' FREQ ', FREQ(IROT,IMODE,ICOORD)
 710  CONTINUE
      WRITE(NUNIT(NCOUNT),*)

C*********************************************************************C
C* TABLES FOR CLASSES AND NODES *C
C*  NODE NUMBERING *C

      WRITE(NUNIT(NCOUNT),*)' INODE, IROT, NODE_LOC(INODE)'
      WRITE(NUNIT(NCOUNT),*)

      DO 650 IROT= 1,NROT

      WRITE(NUNIT(NCOUNT),*) ' BEARINGS NUMBERING '
C* NODE NUMBERING FOR BEARINGS
      CLASS_TABLE(IROT,BEAR)= NBEAR(IROT)
      DO 660 IBEAR=1,NBEAR(IROT)
      NODE_TABLE(IROT,BEAR,IBEAR)= INODE
      CLASS_NODE(INODE)= IROT
      NODE_LOC(INODE)= LBEAR(IROT,IBEAR)
      WRITE(NUNIT(NCOUNT),*) INODE, IROT, NODE_LOC(INODE)
      INODE= INODE + 1
 660  CONTINUE

      WRITE(NUNIT(NCOUNT),*) ' GEARS NUMBERING '
C* NODE NUMBERING FOR GEARS
      CLASS_TABLE(IROT,GEAR)= NGEAR(IROT)
      DO 665 IGEAR=1,NGEAR(IROT)
      NODE_TABLE(IROT,GEAR,IGEAR)= INODE
      CLASS_NODE(INODE)= IROT
      NODE_LOC(INODE)= LGEAR(IROT,IGEAR)
      WRITE(NUNIT(NCOUNT),*) INODE, IROT, NODE_LOC(INODE)
      INODE= INODE + 1
 665  CONTINUE

      WRITE(NUNIT(NCOUNT),*) ' UNBALANCE NUMBERING '
C* NODE NUMBERING FOR UNBALANCE
      CLASS_TABLE(IROT,UNBAL)= NUNBAL(IROT)
      DO 670 IUNB=1, NUNBAL(IROT)
      NODE_TABLE(IROT,UNBAL,IUNB)= INODE
      CLASS_NODE(INODE)= IROT
      NODE_LOC(INODE)= ISTAT_UN(IROT,IUNB)
```

```
      AMPL_PHASE(INODE,1)= SQRT( UNBX(IROT,IUNB)*
     &                           UNBX(IROT,IUNB) +
     &           UNBY(IROT,IUNB)*UNBY(IROT,IUNB) )
      AMPL_PHASE(INODE,2)= ATAN2(UNBY(IROT,IUNB),UNBX(IROT,IUNB))
      AMPL_PHASE(INODE,3)= PHIADD(IROT,IUNB)

      WRITE(NUNIT(NCOUNT),*) INODE, IROT, NODE_LOC(INODE)
      INODE= INODE + 1
  670 CONTINUE

  650 CONTINUE
      NNODE= INODE-1

C* CHECK
      NNODE= INODE-1
      DO 222 INODE2=1,NNODE
      WRITE(NUNIT(NCOUNT),223) ( AMPL_PHASE(INODE2,I),I=1,3 )
  223 FORMAT( 1X, 3( E15.7,1X ) )
  222 CONTINUE

C* CAN BE MODERNIZE TO RESEMBLE CASING FROM ROTOR

      DO 652 IROT=1,NROT
      CLASS_NAME(IROT)= ROTOR
  652 CONTINUE

      WRITE(NUNIT(NCOUNT),*)
      WRITE(NUNIT(NCOUNT),*) ' NNODE = ', NNODE
C* MODE SHAPES

      DO 620 INODE=1,NNODE
      IROT= CLASS_NODE(INODE)
      ILOC= NODE_LOC(INODE)
      DO 620 IMODE=1,NMODE(IROT)
      SHAPE_NODE(INODE,IMODE,1)= SHAPE(IROT,IMODE,ILOC,1)
      SHAPE_NODE(INODE,IMODE,2)= SHAPE(IROT,IMODE,ILOC,2)
      SHAPE_NODE(INODE,IMODE,3)= SHAPE(IROT,IMODE,ILOC,1)
      SHAPE_NODE(INODE,IMODE,4)= SHAPE(IROT,IMODE,ILOC,2)
      SHAPE_NODE(INODE,IMODE,5)= SHAPE(IROT,IMODE,ILOC,3)
      SHAPE_NODE(INODE,IMODE,6)= SHAPE(IROT,IMODE,ILOC,4)

      WRITE(NUNIT(NCOUNT),*) ' INODE ', INODE
      WRITE(NUNIT(NCOUNT),622) (SHAPE_NODE(INODE,IMODE,I),I=1,6)
  620 CONTINUE

      WRITE(NUNIT(NCOUNT),*)
      WRITE(NUNIT(NCOUNT),*) ' INODE   IROT    ILOC '
      DO 630 INODE=1,NNODE
      IROT= CLASS_NODE(INODE)
      ILOC= NODE_LOC(INODE)
      WRITE(NUNIT(NCOUNT),*) INODE,' ',IROT,' ',ILOC
  630 CONTINUE
C*************************************************************C
C* NODE CONNECTIONS FOR BEARINGS *C
      WRITE(NUNIT(NCOUNT),*)
      WRITE(NUNIT(NCOUNT),*) ' NODE CONNECTIONS '
      WRITE(NUNIT(NCOUNT),*) ' INODE JNODE ICON '
      ICON=1

      DO 621 IROT=1,NROT
C* NODE CONNECTIONS FOR BEARINGS
      DO 621 IBEAR=1,NBEAR(IROT)
      DO 629 JROT=1,NROT
      IF( JROT .GT. IROT ) GOTO 629
```

```
      DO 629 JBEAR=1,NBEAR(JROT)

      IF( NCON_BEAR(IROT,IBEAR,JROT,JBEAR) .EQ. 1 ) THEN
      INODE= NODE_TABLE(IROT,IBEAR,IBEAR)
      NODE_CON(INODE,1)= 1
      JNODE= NODE_TABLE(JROT,JBEAR,JBEAR)
      NODE_CON(INODE,2)= JNODE
      NODE_CON(INODE,3)= ICON
      WRITE(NUNIT(NCOUNT),*)  INODE, JNODE, ICON
 * SYMMETRY OF THE CONNECTIONS
      NODE_CON(JNODE,1)= 1
      NODE_CON(JNODE,2)= INODE
      NODE_CON(JNODE,3)= ICON
      WRITE(NUNIT(NCOUNT),*)  JNODE, INODE, ICON
      DO 624 I=1,6
      DO 624 J=1,6
      STIF_BEAR(ICON,I,J)= STIF2(IROT,IBEAR,I,J)
      DAMP_BEAR(ICON,I,J)= DAMP2(IROT,IBEAR,I,J)
      STIF_AV(ICON,I,J)= STIFA(IROT,IBEAR,I,J)
 624  CONTINUE
      ICON= ICON + 1
      ENDIF
C* DEVELOP CLEAR CONNECTION TABLE, HERE 2= GROUND
      IF( NCON_BEAR(IROT,IBEAR,JROT,JBEAR) .EQ. 2 ) THEN
      INODE= NODE_TABLE(IROT,BEAR,IBEAR)
      NODE_CON(INODE,1)= 1
C* GROUND CONNECTION
      JNODE= NNODE+1
      NODE_TABLE(NROT+1,BEAR,1)= JNODE
      CLASS_NODE(JNODE)= NROT+1
      CLASS_NAME(NROT+1)= GROUND

      NODE_CON(INODE,2)= JNODE
      NODE_CON(INODE,3)= ICON
      WRITE(NUNIT(NCOUNT),*)  INODE, JNODE, ICON
      DO 625 I=1,6
      DO 625 J=1,6
      STIF_BEAR(ICON,I,J)= STIF2(IROT,IBEAR,I,J)
      DAMP_BEAR(ICON,I,J)= DAMP2(IROT,IBEAR,I,J)
      STIF_AV(ICON,I,J)= STIFA(IROT,IBEAR,I,J)
625   CONTINUE
      ICON= ICON + 1
      ENDIF

629   CONTINUE
 621  CONTINUE
      NCON= ICON-1

C*************************************************************C
C* CHECK
      WRITE(NUNIT(NCOUNT),*) ' BEARINGS NCON = ', NCON
      WRITE(NUNIT(NCOUNT),*)
      WRITE(NUNIT(NCOUNT),*)' STIF_BEAR(IROT,I,J) '
      DO 71 ICON=1,NCON
      WRITE(NUNIT(NCOUNT),72)((STIF_BEAR(ICON,I,J), I=1,6),J=1,6)
71    CONTINUE
      WRITE(NUNIT(NCOUNT),*)

      WRITE(NUNIT(NCOUNT),*)
      WRITE(NUNIT(NCOUNT),*)' STIF_AV(IROT,I,J) '
      DO 73 ICON=1,NCON
      WRITE(NUNIT(NCOUNT),72)((STIF_AV(ICON,I,J), I=1,6),J=1,6)
73    CONTINUE
      WRITE(NUNIT(NCOUNT),*)

      WRITE(NUNIT(NCOUNT),*)
```

```
      WRITE(NUNIT(NCOUNT),*)' DAMP_BEAR(IROT,I,J) '
      DO 74 ICON=1,NCON
      WRITE(NUNIT(NCOUNT),72)((DAMP_BEAR(ICON,I,J), I=1,6),J=1,6)
 74   CONTINUE
      WRITE(NUNIT(NCOUNT),*)

 72   FORMAT(1X, 6(E10.4,1X) )
C*
C*********************************************C
C* CHECK
      WRITE(NUNIT(NCOUNT),*) ' MODE SHAPES FOR ALL NODES '
      WRITE(NUNIT(NCOUNT),*) ' NNODE = ', NNODE
      DO 627 INODE=1,NNODE
      DO 627 IMODE=1,NMODE(1)
      WRITE(NUNIT(NCOUNT),622)(SHAPE_NODE(INODE,IMODE,I),I=1,6)
 622  FORMAT(1X, 6(F11.5) )
 627  CONTINUE
      WRITE(NUNIT(NCOUNT),*)

C*************************************************************C
C*********************************************C
C* NODE CONNECTIONS FOR GEARS IN THE SYSTEM *C

      ICON= NCON + 1
      DO 750 IROT= 1,NROT
      DO 750 IGEAR= 1,NGEAR(IROT)
      DO 755 JROT= 1,NROT
      IF( JROT .EQ. IROT ) GOTO 755
      DO 755 JGEAR= 1,NGEAR(JROT)

      IF( NCON_GEAR(IROT,IGEAR,JROT,JGEAR) .NE. 0 ) THEN
      INODE= NODE_TABLE(IROT,GEAR,IGEAR)
C* FIX IT LATER= MAKE MULTIPLE CONNECTIONS EASY
      NODE_CON(INODE,1)=1
      JNODE= NODE_TABLE(JROT,GEAR,JGEAR)
      ITYPE= NCON_GEAR(IROT,IGEAR,JROT,JGEAR)
      NODE_CON(INODE,2)= JNODE
      NODE_CON(INODE,3)= ICON
      NODE_CON(INODE,4)= ITYPE
       NODE_CON(INODE,5)= ITYPE
C* SYMMETRY OF THE CONNECTIONS
      NODE_CON(JNODE,1)=1
      NODE_CON(JNODE,2)=INODE
      NODE_CON(JNODE,3)=ICON
      NODE_CON(JNODE,4)= ITYPE
       NODE_CON(JNODE,5)= ITYPE
      WRITE(NUNIT(NCOUNT),*) ' GEARS CONNECTION '
      WRITE(NUNIT(NCOUNT),*) INODE, JNODE, ICON, ITYPE

      ICON= ICON + 1
      ENDIF
 755  CONTINUE
 750  CONTINUE
      NCON2= ICON-1-NCON

      WRITE(NUNIT(NCOUNT),*)
      WRITE(NUNIT(NCOUNT),*) 'THERE ARE ',NCON2,' GEAR CONNECTIONS'
      WRITE(NUNIT(NCOUNT),*)

C*************************************C
C* MODAL EXTERNAL TORQUE CALCULATION   *C
C*************************************C

      DO 760 IROT=1,NROT
      INODE= NODE_TABLE(IROT,EXTERN,1)
```

```
      DO 760 IMODE=1,NMODE(IROT)
      FEXT(IROT,IMODE,6)= TORQUE(IROT)*SHAPE_NODE(INODE,IMODE,6)
  760 CONTINUE

C*********************************C
C* FILE OUTPUT NUMBERING *C
      K=1
      DO 24 IROT=1,NROT
      NNONE= CLASS_TABLE(IROT,NONE)
      DO 24 INONE= 1,NNONE
      NUNIT_OUT(IROT,INONE)= K
      K= K + 1
   24 CONTINUE

C*****************************C
      DO 205 IROT=1,NROT
      TCYCLE(IROT)=60.0/ANGULA(IROT,1)
  205 CONTINUE
C*****************************C
      TIME=0.0
      NCALC= NCYCLE*NPOINTS
      IF( NCALC .EQ. 0 ) STOP
      NCYCLE_OUT=NCYCLE-NCYCLE_TRAN
      NCALC_OUT= (NCYCLE-NCYCLE_TRAN)*NPOINTS
      NTOTAL= NCYCLE*NPOINTS*NFACT
C* GOOD CONVERGENCE IS ABOUT DELTAT=10E-6    *C
      FSAMPLE= ANGULA(1,1)*NPOINTS/60.0
 4000 CONTINUE
      DELTAT= TCYCLE(1)/(NPOINTS*NFACT)

C      IF( DELTAT .GT. 1E-6 ) THEN
C      NFACT= NFACT*2
C      GOTO 4000
C      ENDIF

      WRITE(NUNIT(NCOUNT),*) 'DELTAT= ',DELTAT,' NFACT= ',NFACT

      OPEN(40,FILE='head.out',STATUS='NEW')
      WRITE(40,*) NCYCLE_OUT,NCALC_OUT,
     &                    ANGULA(1,1) ,FSAMPLE
      OPEN(41,FILE='head2.out',STATUS='NEW')
      WRITE(41,*) NCYCLE_OUT,NCALC_OUT,
     &            ANGULA(1,1), NPOINTS

C****** MAIN LOOP ***************C
      DO 100 ICALC= 1,NCALC
      DO 105 IFACT = 1,NFACT
      DO 110 IROT=1,NROT

CC      IF( ICALC .GT. 2 ) STOP
CC      PRINT *,'BEFORE BEARF'
      CALL BEARF( IROT, NODE_TABLE, CLASS_TABLE, NODE_CON,
     & CLASS_NODE, NMODE,
     & SHAPE_NODE, STIF_BEAR, STIF_AV, DAMP_BEAR,
     & XN, VN, FBEAR )

CC      PRINT *,'BEFORE UNBF'
      CALL UNBF( IROT, NODE_TABLE, CLASS_TABLE, NMODE,
     & SHAPE_NODE, ANGULA, AMPL_PHASE, TIME, FUNB )

CC      PRINT *,' BEFORE GEARF '
      ACUR(IROT)= 360.0*TIME/TCYCLE(IROT)
      CALL GEARF( IROT, NODE_TABLE, CLASS_TABLE, NODE_CON,
     & CLASS_NODE, NMODE,
     & SHAPE_NODE, XN, VN, FGEAR,
     & AGEAR, ACUR, STIF_GEAR, NPOINT, AGSTIF, TMR,
```

```
     & GEAR_STIF,ITYPE_DAM )

C      PRINT *,'BEFORE EQU'
       CALL EQU( IROT, NMODE, FREQ, FUNB, FBEAR, FGEAR,
     &           XN, AN )

       DO 210 IMODE= 1, NMODE(IROT)
       DO 210 ICOORD=1,6
       VNP(IROT,IMODE,ICOORD)= VN(IROT,IMODE,ICOORD)
       XNP(IROT,IMODE,ICOORD)= XN(IROT,IMODE,ICOORD)
 210   CONTINUE

       CALL EULER( IROT, NMODE, DELTAT, XNP, VNP, AN )

C      PRINT *,'BEFORE BEARF2'
       CALL BEARF( IROT, NODE_TABLE, CLASS_TABLE, NODE_CON,
     & CLASS_NODE, NMODE,
     & SHAPE_NODE, STIF_BEAR, STIF_AV, DAMP_BEAR,
     & XNP, VNP, FBEAR )

C      PRINT *,'BEFORE UNBF2'
       CALL UNBF( IROT, NODE_TABLE, CLASS_TABLE, NMODE,
     & SHAPE_NODE, ANGULA, AMPL_PHASE, TIME+DELTAT, FUNB )

C      PRINT *,' BEFORE GEARF2 '
       ACUR(IROT)= 360.0*(TIME+DELTAT)/TCYCLE(IROT)
       CALL GEARF( IROT, NODE_TABLE, CLASS_TABLE, NODE_CON,
     & CLASS_NODE, NMODE,
     & SHAPE_NODE, XNP, VNP, FGEAR,
     & AGEAR, ACUR, STIF_GEAR, NPOINT, AGSTIF, TMR,
     & GEAR_STIF,ITYPE_DAM )

       CALL EQU( IROT, NMODE, FREQ, FUNB, FBEAR, FGEAR,
     &           FEXT, XNP, ANP )

       CALL NEWMARK( IROT, NMODE, DELTAT, XN, VN, AN, ANP )

       DO 200 IMODE=1,NMODE(IROT)
       DO 200 ICOORD=1,6
       AN(IROT,IMODE,ICOORD)=0.0
       ANP(IROT,IMODE,ICOORD)=0.0
 200   CONTINUE

 110   CONTINUE
       TIME=TIME + DELTAT
 105   CONTINUE

       IF ( ICALC .LE. NCYCLE_TRAN*NPOINTS ) GOTO 100

       CALL OUTP(NUNIT,NCOUNT,NUNIT_OUT,
     &   NROT,NMODE, CLASS_TABLE,
     &   NODE_TABLE, SHAPE_NODE, XN,VN )

C       WRITE(66,*) GEAR_STIF

 100   CONTINUE
C*****************************************************************C
       RETURN
       END

C*****************************************************************C
       SUBROUTINE BEARF( IROT, NODE_TABLE, CLASS_TABLE, NODE_CON,
     & CLASS_NODE, NMODE,
     & SHAPE_NODE, STIF_BEAR, STIF_AV, DAMP_BEAR, XN, VN, FBEAR )

       PARAMETER NONE= 1
```

```
      PARAMETER UNBAL= 2
      PARAMETER BEAR= 3
      PARAMETER GEAR= 4

      PARAMETER N20=20
      PARAMETER N50=50

C* GLOBAL VARIABLES *C
      INTEGER NODE_TABLE(N20,5,N50)
      INTEGER CLASS_TABLE(N20,N20)
      INTEGER NODE_CON(N50,N50)
      INTEGER CLASS_NODE(N50)
      INTEGER NMODE(N50)

      REAL SHAPE_NODE(N50,N20,6)
      REAL STIF_BEAR(N20,6,6), STIF_AV(N20,6,6)
      REAL DAMP_BEAR(N20,6,6)
      REAL XN(N20,N20,6), VN(N20,N20,6)
      REAL FBEAR(N20,N20,6), FB(6)

C* LOCAL VARIABLES *C
      INTEGER IBEAR, NBEAR, IMODE, ICOORD

      REAL X1(6), V1(6), X2(6), V2(6)

C* INITIALIZE VARIABLES *C

      DO 17 IMODE=1,NMODE(IROT)
      DO 17 ICOORD=1,6
      FBEAR(IROT,IMODE,ICOORD)= 0.0
 17   CONTINUE

C* START CALCULATIONS *C

C      WRITE(50,*) ' IN BEAR '

      NBEAR= CLASS_TABLE(IROT,BEAR)

      DO 100 IBEAR= 1, NBEAR

      DO 10 ICOORD=1,6
      X1(ICOORD)= 0.0
      V1(ICOORD)= 0.0
 10   CONTINUE

      INODE= NODE_TABLE(IROT,BEAR,IBEAR)

C      WRITE(50,*) ' INODE ', INODE

      DO 110 IMODE=1,NMODE(IROT)
      DO 110 ICOORD= 1,6
      X1(ICOORD)= X1(ICOORD) +
     & XN(IROT,IMODE,ICOORD)*SHAPE_NODE(INODE,IMODE,ICOORD)
      V1(ICOORD)= V1(ICOORD) +
     & VN(IROT,IMODE,ICOORD)*SHAPE_NODE(INODE,IMODE,ICOORD)
110   CONTINUE

      NCON= NODE_CON(INODE,1)

C      WRITE(50,*) ' NCON ', NCON

C* LOOP FOR ALL CONNECTIONS *C

      DO 120 ICON= 2, 2*NCON+1, 2

      JNODE= NODE_CON(INODE,ICON)
```

```
      JCON= NODE_CON(INODE,ICON+1)
      JROT= CLASS_NODE(JNODE)

C      WRITE(50,*) ' IROT  INODE  JNODE  JCON  JROT '
C      WRITE(50,*)   IROT, INODE, JNODE, JCON, JROT

C*? FIND CLASS OF THE NODE *C
      DO 20 ICOORD=1,6
      X2(ICOORD)= 0.0
      V2(ICOORD)= 0.0
      FB(ICOORD)=0.0
 20   CONTINUE

      DO 130 IMODE= 1,NMODE(JROT)
      DO 130 ICOORD= 1,6
      X2(ICOORD)= X2(ICOORD) +
     & XN(JROT,IMODE,ICOORD)*SHAPE_NODE(JNODE,IMODE,ICOORD)
      V2(ICOORD)= V2(ICOORD) +
     & VN(JROT,IMODE,ICOORD)*SHAPE_NODE(JNODE,IMODE,ICOORD)
 130  CONTINUE

      DO 150 ICOORD= 1,6
      SUMX=0.0
      SUMV=0.0
      SUMA=0.0
      DO 160 JCOORD= 1,6
      SUMX= SUMX +
     & STIF_BEAR(JCON,ICOORD,JCOORD)*( X1(JCOORD) - X2(JCOORD) )
      SUMV= SUMV +
     & DAMP_BEAR(JCON,ICOORD,JCOORD)*( V1(JCOORD) - V2(JCOORD) )
      SUMA= SUMA +
     & STIF_AV(JCON,ICOORD,JCOORD)*X1(JCOORD)
 160  CONTINUE
      FB(ICOORD)=  SUMV + SUMX - SUMA
 150  CONTINUE

C      WRITE(50,*) ' FB  1 2 3 4 '
C      WRITE(50,*)   FB(1), FB(2), FB(3), FB(4)

      DO 170 IMODE=1,NMODE(IROT)

      FBEAR(IROT,IMODE,1)= FBEAR(IROT,IMODE,1) +
     &          SHAPE_NODE(INODE,IMODE,1)*FB(1) +
     &          SHAPE_NODE(INODE,IMODE,2)*FB(2)
      FBEAR(IROT,IMODE,2)= FBEAR(IROT,IMODE,1)

      FBEAR(IROT,IMODE,3)= FBEAR(IROT,IMODE,3) +
     &          SHAPE_NODE(INODE,IMODE,3)*FB(3) +
     &          SHAPE_NODE(INODE,IMODE,4)*FB(4)
      FBEAR(IROT,IMODE,4)= FBEAR(IROT,IMODE,3)

      FBEAR(IROT,IMODE,5)= FBEAR(IROT,IMODE,5) +
     &          SHAPE_NODE(INODE,IMODE,5)*FB(5)

      FBEAR(IROT,IMODE,6)= FBEAR(IROT,IMODE,6) +
     &          SHAPE_NODE(INODE,IMODE,6)*FB(6)

 170  CONTINUE

 120  CONTINUE

 100  CONTINUE
C*****************************************************************C
      RETURN
      END
```

```
C*********************************************************************C
C*********************************************************************C
      SUBROUTINE UNBF( IROT, NODE_TABLE, CLASS_TABLE,
     & NMODE,
     & SHAPE_NODE, ANGULA, AMPL_PHASE, TIME, FUNB )

      PARAMETER NONE= 1
      PARAMETER UNBAL= 2
      PARAMETER BEAR= 3
      PARAMETER GEAR= 4

      PARAMETER N20=20
      PARAMETER N50=50

      PARAMETER ID_X=1
      PARAMETER ID_Y=3
      PARAMETER PI=3.1415926

C* GLOBAL VARIABLES *C
      INTEGER NODE_TABLE(N20,5,N50)
      INTEGER CLASS_TABLE(N20,N20)
      INTEGER NMODE(N50)

      REAL SHAPE_NODE(N50,N20,6)
      REAL FUNB(N20,N20,6)
      REAL TIME

      REAL AMPL_PHASE(N50,3)
      REAL ANGULA(N20,2)

C* LOCAL VARIABLES *C
      INTEGER IUNB, NUNB, IMODE, ICOORD

      REAL AMPL, PHASE, PHIADD
      REAL ARG
      REAL SPEED, ANGACL
      REAL OMEGA

C*************************C


      SPEED= ANGULA(IROT,1)
      ANGACL= ANGULA(IROT,2)

      OMEGA=PI*SPEED/30.0

      DO 10 IMODE=1,NMODE(IROT)
      DO 10 ICOORD=1,6
      FUNB(IROT,IMODE,ICOORD)=0.0
      FUNB(IROT,IMODE,ICOORD)=0.0
10    CONTINUE

      NUNB= CLASS_TABLE(IROT,UNBAL)

C       PRINT *, NUNB

      DO 20 IUNB=1,NUNB

      INODE= NODE_TABLE(IROT,UNBAL,IUNB)

CC      PRINT *, INODE

      AMPL= AMPL_PHASE(INODE,1)
      PHASE= AMPL_PHASE(INODE,2)
      PHIADD= AMPL_PHASE(INODE,3)
      ARG= OMEGA*TIME + PHASE + PHIADD
```

```
CC       PRINT *, AMPL, PHASE, ARG, OMEGA, TIME

       DO 30 IMODE=1,NMODE(IROT)

        FUNB(IROT,IMODE,ID_X)=
      &                 AMPL*SHAPE_NODE(INODE,IMODE,ID_X)*
      &               ( ANGACL*SIN(ARG) + OMEGA*OMEGA*COS(ARG) )
        FUNB(IROT,IMODE,ID_Y)=
      &                 AMPL*SHAPE_NODE(INODE,IMODE,ID_Y)*
      &               ( ANGACL*COS(ARG) + OMEGA*OMEGA*SIN(ARG) )
CC       PRINT *, SHAPE_NODE(INODE,IMODE,ID_X), FUNB(IROT,IMODE,ID_X)
 30    CONTINUE

 20    CONTINUE

C*********************************************************************C
       RETURN
       END
C*********************************************************************C

C*********************************************************************C
C**********************  SUBROUTINE EQUATION **************************C
C*********************************************************************C
       SUBROUTINE EQU( IROT, NMODE, FREQ, FUNB, FBEAR, FGEAR,
      &                FEXT, XN, AN )

       PARAMETER N10= 10
       PARAMETER N20= 20
       PARAMETER N50=50

C* GLOBAL VARIABLES *C

       INTEGER IROT
       INTEGER NMODE(N50)

       REAL FREQ(N10,N10,N10)
       REAL FUNB(N20,N20,6)
       REAL FBEAR(N20,N20,6)
       REAL FGEAR(N20,N20,6)
       REAL FEXT(N20,N20,6)
       REAL XN(N20,N20,6)
       REAL AN(N20,N20,6)

C* LOCAL VARIABLES *C
       INTEGER ICOORD, IMODE

C************* THE EQUATIONS OF MOTION  ****************************C

C       WRITE(50,*) ' IN EQU '

       DO 100 IMODE=1,NMODE(IROT)

       AN(IROT,IMODE,1)= FUNB(IROT,IMODE,1) -
      &                  FBEAR(IROT,IMODE,1) -
      &    FREQ(IROT,IMODE,1)**2*XN(IROT,IMODE,1) +
      &                  FGEAR(IROT,IMODE,1)

       AN(IROT,IMODE,2)= AN(IROT,IMODE,1)

       AN(IROT,IMODE,3)= FUNB(IROT,IMODE,3) -
      &                  FBEAR(IROT,IMODE,3) -
      &    FREQ(IROT,IMODE,3)**2*XN(IROT,IMODE,3) +
      &                  FGEAR(IROT,IMODE,3)

       AN(IROT,IMODE,4)= AN(IROT,IMODE,3)
```

```
      AN(IROT,IMODE,5)= -FBEAR(IROT,IMODE,5) -
     &    FREQ(IROT,IMODE,5)**2*XN(IROT,IMODE,5) +
     &                    FGEAR(IROT,IMODE,5)

      AN(IROT,IMODE,6)= -FBEAR(IROT,IMODE,6) -
     &    FREQ(IROT,IMODE,6)**2*XN(IROT,IMODE,6) +
     &                    FGEAR(IROT,IMODE,6)      +
     &                    FEXT(IROT,IMODE,6)

       WRITE(50,*) ' IROT IMODE ICOORD  FUNB FBEAR '
C      WRITE(50,*) IROT,IMODE
C      WRITE(50,*) FUNB(IROT,IMODE,1), -FBEAR(IROT,IMODE,1)

100   CONTINUE

C*****************************************************************C
      RETURN
      END
C*****************************************************************C
      SUBROUTINE EULER( IROT, NMODE, DELTAT, XN, VN, AN )

      PARAMETER N20=20
      PARAMETER N50=50

      INTEGER NMODE(N50)

      REAL H,DELTAT

      REAL XN(N20,N20,6), VN(N20,N20,6), AN(N20,N20,6)
      REAL XNP(N20,N20,6), VNP(N20,N20,6)

      INTEGER IMODE, ICOORD

C***************    THE EULER METHOD    ******************************C

      H=DELTAT

      DO 10 IMODE= 1,NMODE(IROT)
      DO 20 ICOORD=1,6
      VNP(IROT,IMODE,ICOORD)= VN(IROT,IMODE,ICOORD)  +
     &                        H*AN(IROT,IMODE,ICOORD)
      XNP(IROT,IMODE,ICOORD)= XN(IROT,IMODE,ICOORD) +
     &  0.5*H*( VNP(IROT,IMODE,ICOORD) + VN(IROT,IMODE,ICOORD) )
20    CONTINUE
10    CONTINUE

      DO 30 IMODE= 1,NMODE(IROT)
      DO 30 ICOORD=1,6
      VN(IROT,IMODE,ICOORD)= VNP(IROT,IMODE,ICOORD)
      XN(IROT,IMODE,ICOORD)= XNP(IROT,IMODE,ICOORD)
30    CONTINUE

      RETURN
      END
C*****************************************************************C
C*****************************************************************C
      SUBROUTINE NEWMARK( IROT, NMODE, DELTAT, XN, VN, AN, ANP )

      PARAMETER N20=20
      PARAMETER N50=50

      INTEGER NMODE(N50)

      REAL H,DELTAT
```

```
      REAL XN(N20,N20,6), VN(N20,N20,6), AN(N20,N20,6)
      REAL ANP(N20,N20,6)

      INTEGER IMODE, ICOORD
      REAL BETA

      BETA=0.167

C******* NEWMARK-BETA METHOD ***************************************C
      H=DELTAT

      DO 20 IMODE= 1,NMODE(IROT)
      DO 20 ICOORD=1,6

      XN(IROT,IMODE,ICOORD)=XN(IROT,IMODE,ICOORD) +
     &                      VN(IROT,IMODE,ICOORD)*H +
     & (0.5 - BETA )*AN(IROT,IMODE,ICOORD)*H*H +
     &         BETA*ANP(IROT,IMODE,ICOORD)*H*H
      VN(IROT,IMODE,ICOORD)=VN(IROT,IMODE,ICOORD) +
     & 0.5*H*( AN(IROT,IMODE,ICOORD)+ANP(IROT,IMODE,ICOORD) )

 20   CONTINUE

      RETURN
      END
C*****************************************************************C
C*****************************************************C
C*************** SUBROUTINE OUTPUT ********************C
C***  OUTPUT AND POST-PROCESSING OF THE RESULTS   ******C
      SUBROUTINE OUTP(NUNIT,NCOUNT,NUNIT_OUT,
     &  NROT,NMODE, CLASS_TABLE,
     &  NODE_TABLE, SHAPE_NODE, XN,VN )

      PARAMETER NONE= 1
      PARAMETER UNBAL= 2
      PARAMETER BEAR= 3
      PARAMETER GEAR= 4

      PARAMETER N10=10
      PARAMETER N20=20
      PARAMETER N50=50
      PARAMETER SC=10000

      INTEGER NUNIT(N50)
      INTEGER NUNIT_OUT(N20,N20)
      INTEGER NMODE(N50)
      INTEGER JSTAT, IUNIT
      INTEGER NROT, IROT, INODE, IMODE
      INTEGER NODE_TABLE(N20,5,N50)
      INTEGER CLASS_TABLE(N20,N20)

      REAL SHAPE_NODE(N50,N20,6)
      REAL XN(N20,N20,6), VN(N20,N20,6)
      REAL XD(6), VD(6)

      IROT=0
      IUNIT=0

      DO 10 IROT=1,NROT

      NNONE= CLASS_TABLE(IROT,NONE)
      DO 10 INONE=1,NNONE

      DO 30 ICOORD=1,6
      XD(ICOORD)=0.0
      VD(ICOORD)=0.0
```

```
 30    CONTINUE

       INODE= NODE_TABLE(IROT,NONE,INONE)

       DO 20 IMODE=1,NMODE(IROT)
       DO 20 ICOORD=1,6
       XD(ICOORD)= XD(ICOORD) +
      &     XN(IROT,IMODE,ICOORD)*SHAPE_NODE(INODE,IMODE,ICOORD)
       VD(ICOORD)= VD(ICOORD) +
      &     VN(IROT,IMODE,ICOORD)*SHAPE_NODE(INODE,IMODE,ICOORD)
 20    CONTINUE

       IUNIT= NUNIT_OUT(IROT,INONE)
       WRITE(NUNIT(NCOUNT+IUNIT),*) XD(3)

 100   FORMAT( 1X, 2(G15.9,1X) )

 10    CONTINUE

       RETURN
       END
C*********************************************************************C
C*********************************************************************C
C* CODE FOR GEAR FORCE SUBROUTINE
C*********************************************************************C
       SUBROUTINE GEARF(IROT, NODE_TABLE, CLASS_TABLE, NODE_CON,
      & CLASS_NODE, NMODE,
      & SHAPE_NODE, XN, VN, FGEAR,
      & AGEAR, ACUR, STIF_GEAR, NPOINT, AGSTIF, TMR, GEAR_STIF,
      & ITYPE_DAM )

       PARAMETER NONE= 1
       PARAMETER UNBAL= 2
       PARAMETER BEAR= 3
       PARAMETER GEAR= 4

       PARAMETER N10=10
       PARAMETER N20=20
       PARAMETER N50=50
       PARAMETER N120=120

       PARAMETER PI=3.1415926

* GLOBAL VARIABLES *C
       INTEGER NODE_TABLE(N20,5,N50)
       INTEGER CLASS_TABLE(N20,N20)
       INTEGER NODE_CON(N50,N50)
       INTEGER CLASS_NODE(N50)
       INTEGER NMODE(N50)
       INTEGER  NPOINT(N20)
       INTEGER ITYPE_DAM(N50,N50)

       REAL SHAPE_NODE(N50,N20,6)
       REAL XN(N20,N20,6), VN(N20,N20,6)
       REAL FGEAR(N20,N20,6)
       REAL AGEAR(N20,N20,N10), AGSTIF(N20,N120)
       REAL STIF_GEAR(n20,n120)
       REAL TMR(N20,3,3)
       REAL ACUR(N20)

C*********************************************************************C
       INTEGER NGEAR
       INTEGER INODE, JNODE, ICON, JCON, IROT
       INTEGER AXE, RAD

       REAL PHIN, PSI, GAMA, ALPHA
```

```
      REAL SX, SV, FG(6), FX,FY,FZ,FN, DLENGTH
      REAL F_RADIAL, F_TANG, F_AXIAL
      REAL R1, R2
      REAL GEAR_STIF
      REAL DX, DY, DROT

      REAL X1(6), V1(6), X2(6), V2(6)
      REAL X21(6), V21(6)

C*****************************************************************C
C* INITIALIZE VARIABLES *C
C*****************************************************************C
      DO 17 IMODE=1,NMODE(IROT)
      DO 17 ICOORD=1,6
      FGEAR(IROT,IMODE,ICOORD)= 0.0
 17   CONTINUE

C*****************************************************************C
C* START CALCULATIONS *C
C*************************************************C

      NGEAR= CLASS_TABLE(IROT,GEAR)

      DO 100 IGEAR= 1, NGEAR

      DO 10 ICOORD=1,6
      X1(ICOORD)= 0.0
      V1(ICOORD)= 0.0
 10   CONTINUE

C*************************************************C

      R1=   AGEAR(IROT,IGEAR,1)

C* ALL ANGLES IN THE INPUT WERE IN THE DEGREES
C* TRANSFORM ALL ANGLES INTO RADIANS

      DO 11 I=2,5
      AGEAR(IROT,IGEAR,I)= PI*AGEAR(IROT,IGEAR,I)/180.0
 11   CONTINUE

      PHIN= AGEAR(IROT,IGEAR,2)
      PSI=  AGEAR(IROT,IGEAR,3)
      GAMA= AGEAR(IROT,IGEAR,4)
      ALPHA=AGEAR(IROT,IGEAR,5)
C*******************************C

      INODE= NODE_TABLE(IROT,GEAR,IGEAR)

      DO 20 IMODE=1,NMODE(IROT)
      DO 20 ICOORD= 1,6
      X1(ICOORD)= X1(ICOORD) +
     & XN(IROT,IMODE,ICOORD)*SHAPE_NODE(INODE,IMODE,ICOORD)
      V1(ICOORD)= V1(ICOORD) +
     & VN(IROT,IMODE,ICOORD)*SHAPE_NODE(INODE,IMODE,ICOORD)
 20   CONTINUE

      NCON= NODE_CON(INODE,1)

C* LOOP FOR ALL CONNECTIONS *C

      DO 110 ICON= 2, 2*NCON+2, 3

C* FIND CONNECTION

      JNODE= NODE_CON(INODE,ICON)
```

```
      JCON=  NODE_CON(INODE,ICON+1)
      JTYPE= NODE_CON(JNODE,ICON+2)
C       ITYPE_ORG = NODE_CON(JNODE,ICON+3)
      ITYPE= JTYPE
      JROT= CLASS_NODE(JNODE)

C       WRITE(66,*) INODE, JNODE, JCON, ITYPE

C* WORK HERE : CHECK IT AGAIN YOU CAN PUT IT BEFORE BIG LOOP
      NGEAR2= CLASS_TABLE(JROT,GEAR)
      DO 25 JGEAR=1,NGEAR2
      IF( JNODE .EQ. NODE_TABLE(JROT,GEAR,JGEAR) )
     &  R2= AGEAR(JROT,JGEAR,1)
 25   CONTINUE

C* AS A REFERNCE GEAR IS TAKEN THE ONE WITH SMALLER RADIUS

      ANGLE= ACUR(IROT)
      IF( R1 .GT. R2 ) ANGLE= ACUR(JROT)

C*********************************************************************C
C* FIND STIFFNESS FOR THIS CONNECTION
C* WORK HERE!
C*********************************************************************C

      GEAR_STIF=0.0
      NP= NPOINT(ITYPE)

C       WRITE(66,*) 'ITYPE= ',ITYPE,' ANGLE= ', ANGLE
C       PRINT * , 'BEFORE GSTIF'
      CALL GSTIF(ITYPE,NP,AGSTIF,ANGLE,STIF_GEAR,GEAR_STIF,
     &  ITYPE_DAM,JTYPE)
C       IF (ITYPE .NE. JTYPE) ITYPE = JTYPE
C       PRINT * , 'AFTER GSTIF'
CC      WRITE(66,*) 'GEAR_STIF= ', GEAR_STIF
C       IF (ITYPE .NE. JTYPE) NODE_CON(JNODE,ICON+2)=ITYPE
C*********************************************************************C
C* INITIALIZE VARIABLES *C
C*********************************************************************C
      DO 30 ICOORD=1,6
      X2(ICOORD)= 0.0
      X21(ICOORD)= 0.0
      V2(ICOORD)= 0.0
      V21(ICOORD)= 0.0
30    CONTINUE

C*********************************************************************C
C* FIND COORDINATES OF CONNECTED GEAR IN ITS COORD. SYSTEM       *C
C*********************************************************************C

      DO 40 IMODE= 1,NMODE(JROT)
      DO 40 ICOORD= 1,6
      X2(ICOORD)= X2(ICOORD) +
     & XN(JROT,IMODE,ICOORD)*SHAPE_NODE(JNODE,IMODE,ICOORD)
X       V2(ICOORD)= V2(ICOORD) +
X      & VN(JROT,IMODE,ICOORD)*SHAPE_NODE(JNODE,IMODE,ICOORD)
 40   CONTINUE
C*********************************************************************C

C*********************************************************************C
C* COORDINATE TRANSFORMATION OF J-GEAR'S SHAFT COORDINATES INTO  *C
C* I-GEAR'S SHAFT COORDINATES                                    *C
C*********************************************************************C
C* TRANSFORMATION OF X, Y, Z COORDINATES. ANGLES ARE INVARIANT TO
C* COORDINATE TRANSFORMATION.
```

```
      DO 50 ICOORD= 1,6, 2
      SX= 0.0
      SV= 0.0
      DO 51 JCOORD= 1,6, 2
      SX= SX + TMR(JTYPE,ICOORD,JCOORD)*X2(JCOORD)
CC    SV= SV + TMR(JTYPE,ICOORD,JCOORD)*V2(JCOORD)
 51   CONTINUE
      X21(ICOORD)= SX
CC    V21(ICOORD)= SV
 50   CONTINUE
 * ANGLES REMAIN UNCHANGED, I.E. X21(angles coord)= X2(angles coord)
      DO 52 ICOORD= 2,6, 2
      X21(ICOORD)= X2(ICOORD)
 52   CONTINUE

CC      WRITE(66,*) 'DIF ',X2(1)-X21(1), X2(3)-X21(3)

 ***************************************************************C
C* CALCULATE THE SPRING CONTRACTION ( EXPANSION )               *C
C***************************************************************C

      DX= X1(1)-X21(1)
      DY= X1(3)-X21(3)
      DROT= R1*X1(6)-R2*X21(6)

      DLENGTH= ( DX*COS(PHIN) + DY*SIN(PHIN) ) +
     &         DROT*COS(PHIN)

 C      WRITE(66,*) 'IROT ',IROT,' DL= ',DLENGTH
 C      WRITE(66,*) 'DX ', DX, ' DY ', DY, ' DROT ',DROT

CX      DVELOSITY= ( V1(1)-V21(1) )*COS(PHIN) +
 X     &          ( V1(3)-V21(3) )*SIN(PHIN) +
 X     &   ( R1*SPEED(IROT)-R2*SPEED(JROT) )*COS(PHIN)

C***************************************************************C
 * CALCULATE THE GEAR FORCE AT CONTACT POINT                    *C
C***************************************************************C
C GEAR_STIF(JCON)

      FN= -GEAR_STIF*DLENGTH

CC      WRITE(66,*) ' IROT= ',IROT,' FN= ',FN,
 C      WRITE(66,*) ' GEAR_ST ',GEAR_STIF

C***************************************************************C
      FX=0.0
      FY=0.0
      FZ=0.0
      DO 12 I=1,6
 12   FG(I)=0.0

C***************************************************************C
C* PHIN - PRESSURE ANGLE
C* PSI  - HELIX ANGLE
C* GAMA - PITCH ANGLE
C* ALPHA - CONTACT POINT ANGLE

 * F_TANG=     FX
C* F_RADIAL = FY
C* F_AXIAL=    FZ

* CONTACT POINT COORDINATE SYSTEM TRANSFORMATION *C
      F_RADIAL= FN*SIN(PHIN)
      F_TANG=   FN*COS(PHIN)*COS(PSI)
      F_AXIAL=  FN*SIN(PHIN)*SIN(PSI)
```

```
      FX= F_RADIAL
      FY= F_TANG
      FZ= F_AXIAL

C* IN PLANE YZ COORDINATE TRANSFORMATION *C
      AXE= 1
      RAD= 1
      CALL PLANE( FX, FY, FZ, GAMA, AXE, RAD )
C* IN PLANE XY COORDINATE TRANSFORMATION *C
      AXE= 3
      RAD= 1
      CALL PLANE( FX, FY, FZ, ALPHA, AXE, RAD )

      FG(1)= FX
      FG(2)= FG(1)
      FG(3)= FY
      FG(4)= FG(3)
      FG(5)= FZ

C*****************************************************************C

      DO 70 IMODE=1,NMODE(IROT)
      DO 70 ICOORD=1,5,2
      FGEAR(IROT,IMODE,ICOORD)= FGEAR(IROT,IMODE,ICOORD) +
     & SHAPE_NODE(INODE,IMODE,ICOORD)*FG(ICOORD)
 70   CONTINUE

      DO 75 IMODE=1,NMODE(IROT)
      FGEAR(IROT,IMODE,6)= FGEAR(IROT,IMODE,6) +
     & SHAPE_NODE(INODE,IMODE,6)*FY*R1
 75   CONTINUE

110   CONTINUE
100   CONTINUE

C       WRITE(66,*) ' FGEAR= ', FGEAR(IROT,1,1)

      RETURN
      END
C*****************************************************************C
C*****************************************************************C
C*************************************C
C* CODE FOR GEAR STIFFNESS CALCULATION *C
C*************************************C
      SUBROUTINE GSTIF( ITYPE, NPOINT, AGSTIF, ANGLE,
     &                  STIF_GEAR, GEAR_STIF,ITYPE_DAM,JTYPE )

      PARAMETER N10=10
      PARAMETER N20=20
      PARAMETER N50=50
      PARAMETER N120=120

      INTEGER ITYPE, NPOINT, JTYPE
      INTEGER ITYPE_DAM(N50,N50)

      REAL AGSTIF(N20,N120), ANGLE
      REAL STIF_GEAR(n20,n120)
      REAL GEAR_STIF
C*************************C
      INTEGER IPOINT

      REAL A, B, ARATIO, SDIF, AMAX

C** F*** REWRITE WITH FORTRAN FUNCTIONS : MOD & ETC. *C
C       WRITE(66,*) 'ANGLE=  ', ANGLE
```

```
      ITEETH = 1
 10   IF( ANGLE .GE. 360.0 ) THEN
      ANGLE= ANGLE - 360.0
      ITEETH = 1
      GOTO 10
      ENDIF

 20   IF ( ANGLE .GE. 0.0 ) GOTO 25
      ANGLE= ANGLE + 360.0
      ITEETH = 1
      GOTO 20

 25   CONTINUE
      AMAX= AGSTIF(ITYPE,1)
 30   IF ( ANGLE .LE. AMAX ) GOTO 40
      ANGLE= ANGLE - AMAX
      ITEETH = ITEETH +1
      GOTO 30

 40   CONTINUE
CC      IF ( ITEETH .GE. 28) PRINT * , 'ITEETH & ANGLE=  ', ITEETH,ANGLE
      IF ( ITYPE_DAM(JTYPE,ITEETH) .NE. ITYPE) THEN
         ITYPE = ITYPE_DAM(JTYPE,ITEETH)
      END IF
C***************C
       if (iteeth .EQ. 10) then
          write(66,*) 'itype', itype, 'iteeth', iteeth,'jtype', jtype
C         write(66,*) 'angle', angle
          write(66,*) 'stif_gear', (stif_gear(itype, i),i=1,100)
       end if
      GEAR_STIF= STIF_GEAR(ITYPE,1)

C* THERE IS A BUG = FIND & FIX= RUN
      DO 50 IPOINT= 2,NPOINT
      IF( ( ANGLE .GE. AGSTIF(ITYPE, 3+IPOINT) )
     & .AND.
     & ( ANGLE .LT. AGSTIF(ITYPE, 4+IPOINT) ) ) THEN

      A=  ANGLE - AGSTIF(ITYPE,3+IPOINT)
      B=  AGSTIF(ITYPE,4+IPOINT) - AGSTIF(ITYPE,3+IPOINT)
      ARATIO= A/B
      SDIF= STIF_GEAR(ITYPE,IPOINT) - STIF_GEAR(ITYPE,IPOINT-1)
      GEAR_STIF= STIF_GEAR(ITYPE,IPOINT-1) + SDIF*ARATIO

      ENDIF

 50   CONTINUE

CC      WRITE (66,*) 'ITEETH=  ', ITEETH
      ITYPE = JTYPE
      RETURN
      END
C*****************************************************************C
C*****************************************************************C
C* SUBROUTINE FOR IN-PLANE COORDINATE TRANSFORMATION             *C
C*****************************************************************C
      SUBROUTINE PLANE( X, Y, Z, ANGLE, AXE, RAD )
      PARAMETER PI=3.14
      INTEGER AXE, DIR, RAD
      REAL X, Y, Z, ANGLE
      REAL OLD_A, OLD_B, OLD_C
      REAL NEW_A, NEW_B, NEW_C

      IF( RAD .EQ. 0 ) ANGLE= PI*ANGLE/180.0

      DIR= IABS(AXE)/AXE
```

```
      IF( AXE .EQ. 1 ) THEN
      OLD_A= Y
      OLD_B= Z
      OLD_C= X
      ENDIF

      IF( AXE .EQ. 2 ) THEN
      OLD_A= X
      OLD_B= Z
      OLD_C= Y
      ENDIF

      IF( AXE .EQ. 3 ) THEN
      OLD_A= X
      OLD_B= Y
      OLD_C= Z
      ENDIF

      NEW_A= OLD_A*COS(ANGLE) - OLD_B*SIN(ANGLE)
      NEW_B= OLD_A*SIN(ANGLE) + OLD_B*COS(ANGLE)
      NEW_C= DIR*OLD_C

      IF( AXE .EQ. 1 ) THEN
      Y= NEW_A
      Z= NEW_B
      X= NEW_C
      ENDIF

      IF( AXE .EQ. 2 ) THEN
      X= NEW_A
      Z= NEW_B
      Y= NEW_C
      ENDIF

      IF( AXE .EQ. 3 ) THEN
      X= NEW_A
      Y= NEW_B
      Z= NEW_C
      ENDIF

      RETURN
      END
C*****************************************************************C
```

```
C*****************************************************************C
C********************** " THE FINAL INSULT " *********************C
C*****************************************************************C
      SUBROUTINE LAT(NUNIT,NCOUNT,NROT,NSTAT,NBEAR,LBEAR,STIF2,
     &  DAMP2,STIFA,FREQ,SHAPE,NMODE,NCON_BEAR,NSWITCH)
      PARAMETER NROT_SIZE=10
      PARAMETER NSTAT_SIZE=100
      PARAMETER N100=100
      PARAMETER N50=50
      PARAMETER N20=20
      PARAMETER N10=10
      PARAMETER N5=5

      INTEGER NMODE(N50)
      INTEGER NM(N50)
      INTEGER NROT
      INTEGER NUNIT(N20),NSTAT(NROT_SIZE)
      INTEGER NBEAR(NROT_SIZE), LBEAR(NROT_SIZE,10)
      INTEGER NSWITCH(N10)

      REAL SHAPE(N50,N50,N50,4)
      REAL FREQ(N10,N10,N10)
      REAL LMODE(NROT_SIZE,N10,NSTAT_SIZE)
      REAL SLOPE(NROT_SIZE,N10,NSTAT_SIZE)

      INTEGER N, NB, NNCT
      INTEGER NBC
      REAL SPI,SPL,DSP,DDIN
      REAL CRT(N10)
      REAL TCRT(N10,N5)
      REAL  DDPC(N10,N100),EEYTH(N10,N100)
      REAL TDMY(N10,N10,N5),TEEYTH(N10,N100,N5)
      REAL TW(N100,5),TWMOD(N50,N5)

      REAL AKK(N10),TAKK(N10,N5),AKRR(N10),TAKR(N10,N5)

      REAL DEFL(N100), EYTH(N100), WMOD(N100), TMX(N100,N100)
      REAL EY1(N100),  EY2(N100)
      REAL DPC(N100),  EAN1(N100), EAN2(N100)
      REAL KXX(N10), KYY(N10),KRX(N10), KRY(N10)
      REAL TDDPC(N10,N100,N5)
      REAL SHAFT_WEIGHT(NSTAT_SIZE),TOTAL_WEIGHT, W(NSTAT_SIZE)
      REAL EXTERNAL_WEIGHT(NSTAT_SIZE), WEIGHT_STAT(NSTAT_SIZE)
      REAL DOUT(NSTAT_SIZE),DIN(NSTAT_SIZE)
      REAL  ITRANS(NSTAT_SIZE),IPOLAR(NSTAT_SIZE)
      REAL  DX(NSTAT_SIZE),RO(NSTAT_SIZE)
      REAL  EI(NSTAT_SIZE),INERTIA(NSTAT_SIZE)
      REAL  EM(NSTAT_SIZE)
      REAL  STIF_BEAR(NROT_SIZE,N10,6,6)
      REAL  DAMP_BEAR(NROT_SIZE,N10,6,6)

      REAL  STIF2(NROT_SIZE,N10,6,6), STIFA(NROT_SIZE,N10,6,6)
      REAL  DAMP2(NROT_SIZE,N10,6,6)

C*****************************************************************C
      REAL  WJ(NSTAT_SIZE),GJ(NSTAT_SIZE)
      REAL  GGO(NSTAT_SIZE),TK(NSTAT_SIZE)
      REAL  AT(N5,N5)
      REAL FFR(N10),FV(N100,N10),TV(N100,N10),WF(N100,N100)
      REAL FR(N10,N5),FFV(N10,N100,N5)
      REAL ZF(N10,N5),ZFV(N10,N100,N5)
      REAL SP1,SP2,SP3
C*****************************************************************C
C**  FOR SYSTEM NATURAL FREQ. CALCULATIONS      ******************C
      REAL AK,AKR, STIF_AV(N10,N10,4,4)
      REAL FMODE(N10,N10,4), SBARK, SBARC
```

```
      REAL KBAR(N10,N10),CBAR(N10,N10)
      REAL K2BAR(N10,N10),C2BAR(N10,N10)
      REAL AGLOB(10,24,24),ASUPER(N100,N100)
      INTEGER NCON_BEAR(N10,N10,N10,N10)
C************************  INPUT DATA ****************************C
C*****************************************************************C
C************************ START MAIN BODY ************************C

      IROT=1
      READ (NUNIT(1),*)
      READ (NUNIT(1),*) NROT

C********** START ENTERING SHAFT AND BEARINGS DATA ***************C
 10   CONTINUE
      READ (NUNIT(1),*)
      READ (NUNIT(1),*)
      READ (NUNIT(1),*) NSTAT(IROT)
      READ (NUNIT(1),*)
      READ (NUNIT(1),*) ( EXTERNAL_WEIGHT(J),DX(J),DOUT(J),DIN(J),
     +  IPOLAR(J),ITRANS(J),EM(J),RO(J),GGO(J),TK(J), J=1,NSTAT(IROT) )
      READ (NUNIT(1),*)
      READ (NUNIT(1),*)
      READ (NUNIT(1),*) NBEAR(IROT)
      READ (NUNIT(1),*)
      READ (NUNIT(1),*) ( LBEAR(IROT,IBEAR),IBEAR=1,NBEAR(IROT) )
      DO 20 IBEAR=1,NBEAR(IROT)
      READ (NUNIT(1),*)
      READ (NUNIT(1),*)
      READ (NUNIT(1),*)(( STIF_BEAR(IROT,IBEAR,I,J),J=1,6),I=1,6)
      READ (NUNIT(1),*)
      READ (NUNIT(1),*)(( DAMP_BEAR(IROT,IBEAR,I,J),J=1,6),I=1,6)
 20   CONTINUE
      READ (NUNIT(1),*)
      READ (NUNIT(1),*)
      READ (NUNIT(1),*) NMODE(IROT),SPI, SPL, DSP
      READ (NUNIT(1),*)
      READ (NUNIT(1),*) NBC

CCC       NMODE(IROT)=3

C*****************************************************************C
C***  INITIALIZE MODE SHAPES AND FREQUENCIES *********************C
      DO 70 ISPEED=1,NMODE(IROT)
      FREQ(IROT,ISPEED,1)= 0.0
      DO 70 ISTAT=1,NSTAT(IROT)
      SHAPE(IROT,ISPEED,ISTAT,1)= 0.0
      SHAPE(IROT,ISPEED,ISTAT,2)= 0.0
 70   CONTINUE

C*****************************************************************C
C*       * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C*       *   CALCULATE EFFECTIVE INERTIA MOMENT OF ROTORS  *
C*       * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C*****************************************************************C
      PI=3.14159
      E=EM(1)
      DO 30 I=1,NSTAT(IROT)
      INERTIA(I)= PI*( DOUT(I)**4 - DIN(I)**4 )/64
      EI(I)= E*INERTIA(I)
 30   SHAFT_WEIGHT(I)= PI*( DOUT(I)**2 - DIN(I)**2 )*DX(I)*RO(I)/4

      W(1)= SHAFT_WEIGHT(1)/2.0 + EXTERNAL_WEIGHT(1)
      TOTAL_WEIGHT= W(1)
      TOTAL_LENGTH= DX(1)

      DO 40 I=2,NSTAT(IROT)
```

```
      WEIGHT_STAT(I)= SHAFT_WEIGHT(I-1)/2.0 + SHAFT_WEIGHT(I)/2.0
     &              + EXTERNAL_WEIGHT(I)
      W(I)= WEIGHT_STAT(I)
      TOTAL_WEIGHT= TOTAL_WEIGHT + WEIGHT_STAT(I)
      TOTAL_LENGTH= TOTAL_LENGTH + DX(I)
 40   CONTINUE

      IPOLAR(1)= IPOLAR(1) + INERTIA(1)*RO(1)*DX(1)
      ITRANS(1)= ITRANS(1) +
     &  SHAFT_WEIGHT(1)*(( DOUT(1)**2.0 + DIN(1)**2.0 )/16.0 +
     &  (( DX(1)/2.0 )**2.0 )/3.0 )/2.0

      DO 50 I=2,NSTAT(IROT)
      IPOLAR(I)= IPOLAR(I) + RO(I)*INERTIA(I)*DX(I) +
     &             INERTIA(I-1)*DX(I-1)*RO(I-1)
      ITRANS(I)= ITRANS(I) +
     &           SHAFT_WEIGHT(I)*((DOUT(I)**2.0+DIN(I)**2.0)/16.0 +
     &           (( DX(I)/2.0 )**2.0 )/3.0 )/2.0 +
     &     SHAFT_WEIGHT(I-1)*((DOUT(I-1)**2.0+DIN(I-1)**2.0)/16.0 +
     &           (( DX(I-1)/2.0 )**2.0 )/3.0 )/2.0
 50   CONTINUE
C*********************************************************************C
C*************** OUTPUT SHAFT DATA CALCULATIONS ********************C
      WRITE (NUNIT(NCOUNT),*)
      WRITE (NUNIT(NCOUNT),*) '    DATA FOR ROTOR   ', ' #',IROT
      WRITE (NUNIT(NCOUNT),*)
      WRITE (NUNIT(NCOUNT),*)  ' I W(I) DX(I) DOUT(I) DIN(I) INERTIA(I)
     1 IPOLAR(I) ITRANS(I) EI(I) '
      WRITE (NUNIT(NCOUNT),*)
      WRITE (NUNIT(NCOUNT),61) ( I,W(I),DX(I),DOUT(I),DIN(I),INERTIA(I),
     1             IPOLAR(I),ITRANS(I),EI(I) , I=1,NSTAT(IROT) )
      WRITE (NUNIT(NCOUNT),*)
      WRITE (NUNIT(NCOUNT),*) 'TOTAL WEIGHT  ','TOTAL LENGTH '
      WRITE (NUNIT(NCOUNT),*)  TOTAL_WEIGHT ,   TOTAL_LENGTH
      WRITE (NUNIT(NCOUNT),*)
      DO 621 IBEAR=1,NBEAR(IROT)
      KXX(IBEAR)= STIF_BEAR(IROT,IBEAR,1,1)
      KYY(IBEAR)= STIF_BEAR(IROT,IBEAR,3,3)
      KRX(IBEAR)= STIF_BEAR(IROT,IBEAR,2,2)
      KRY(IBEAR)= STIF_BEAR(IROT,IBEAR,4,4)
      WRITE (NUNIT(NCOUNT),*)'BEARING STIFFNESS IN X- AND Y- DIRECTIONS'
      WRITE (NUNIT(NCOUNT),*)  '  KXX  ', '  KYY  '
      WRITE (NUNIT(NCOUNT),*)   KXX(IBEAR),' ',KYY(IBEAR)
      WRITE (NUNIT(NCOUNT),*) ' AVERAGE SPRING BEARING STIFFNESS '
      WRITE (NUNIT(NCOUNT),*)   0.5*(KXX(IBEAR) + KYY(IBEAR))
      WRITE (NUNIT(NCOUNT),*) ' AVERAGE ROTATIONAL BEARING STIFFNESS '
      WRITE (NUNIT(NCOUNT),*)   0.5*(KRX(IBEAR) + KRY(IBEAR))
621   CONTINUE

      WRITE (NUNIT(NCOUNT),*) ' DATA FOR LATERAL MODE CALCULATIONS:'
* SPI=INITIAL SPEED,SPL=FINAL SPEED,DSP=SPEED INCREMENT-RPM  *C
      WRITE (NUNIT(NCOUNT),*) 'SPI',' SPL ',' DSP '
      WRITE (NUNIT(NCOUNT),*) SPI, SPL, DSP
      WRITE (NUNIT(NCOUNT),*) ' TYPE OF BOUNDARY CONDITIONS '
      WRITE (NUNIT(NCOUNT),*) NBC

C*********************  PART TWO ******************************C
C**************************************************************C
C       * * * * * * * * * * * * * * * * * * * * * * * *
C ----- *   CALCULATE ROTOR'S LATERAL MODAL SHAPE    *
C       * * * * * * * * * * * * * * * * * * * * * * * *
C**************************************************************C
C**************************************************C
C* NC=LOCAL CRITICAL SPEED  NO. *C
      NC=1
      MA=0
```

```
      MB=0
      LN=1
      LN=LN+3
      DDIN=DSP
      SPD=SPI
      DETP=0.

C******* START LOOP 290-610 *********C
 290  I=1
      J=1
      SPSQ=SPD*SPD
      ANSP=SPD*0.10471976
      ANSP2=ANSP*ANSP
      VP=0.
      ZMP=0.
      EYP=0.
      ETHP=1.0
      M=1
 300  I=I+1
      II=I-1
      IF (II-LBEAR(IROT,J)) 330,310,330
  10  AK=(KXX(J)+KYY(J))/2.0
      AKR=(KRX(J)+KRY(J))/2.
      AKK(J)=AK
      AKRR(J)=AKR
      TAKK(J,IROT)=AKK(J)
      TAKR(J,IROT)=AKRR(J)
      IF (J-NBEAR(IROT)) 320,340,340
  20  J=J+1
      GO TO 340
 330  AK=0.0
      AKR=0.0
  40  VP=VP+(W(I-1)*ANSP2/386.4-AK)*EYP
      ZMP=ZMP+AKR*ETHP-ANSP2*(ITRANS(I-1))*ETHP/386.4
      EY=EYP+DX(I-1)*ETHP+DX(I-1)**2*ZMP/(2.E6*EI(I-1))+
     1   DX(I-1)**3*VP/(6.E6*EI(I-1))
      ETH=ETHP+DX(I-1)*ZMP/(1.E6*EI(I-1))+
     1    DX(I-1)**2*VP/(2.E6*EI(I-1))
      ZM=ZMP+DX(I-1)*VP
      V=VP
      IF (M.EQ.2) GO TO 350
      EY1(I)=EY
      EAN1(I)=ETH
      IF (I.GT.NSTAT(IROT)) GO TO 360
      ZMP=ZM
      VP=V
      EYP=EY
      ETHP=ETH
      GO TO 300
 350  EY2(I)=EY
      EAN2(I)=ETH
      ZMP=ZM
      VP=V
      EYP=EY
      ETHP=ETH
      IF (I.GT.NSTAT(IROT)) GO TO 370
      GO TO 300
  60  M=2
      ZM1=ZM
      VR1=V
      J=1
      I=1
      EYP=1.
      ZMP=0.
      ETHP=0.
      VP=0.
```

```
      GO TO 300
  370 DET=VR1*ZM-V*ZM1
      IF (ABS(DETP).LT.0.0001) GO TO 420
      IF (MA.EQ.1) GO TO 400
      IF (ABS(DET).LT.1.) GO TO 450
      IF (DETP*DET) 380,420,420
  380 DOLD=DETP
  390 MA=1
      IF (ABS(DET).LT.1.) GO TO 450
      IF (DDIN.LT.1.E-6) GO TO 450
      DDIN=DDIN/2.
      DETPP=DETP
      DETP=DET
      SPD=SPD-DDIN
      GO TO 290
  400 IF (ABS(DET).LT.1.) GO TO 450
      IF (DOLD*DET) 390,420,410
  410 CONTINUE
      IF (ABS(DET).LT.1.) GO TO 450
      IF (DDIN.LT.1.E-6) GO TO 450
      DDIN=DDIN/2.
      SPD=SPD+DDIN
      DETPP=DETP
      DETP=DET
      GO TO 290
  420 IF (LN-54) 440,440,430
  430 CONTINUE
      LN=1
  440 CONTINUE
      LN=LN+1
      SPD=SPD+DSP
      DDIN=DSP
      IF(NC.GT.NMODE(IROT)) GO TO 610
      IF (SPD.GT.SPL) GO TO 610
      DETPP=DETP
      DETP=DET
      SSPD=SPD
      GO TO 290
  450 MA=0
      LN=LN+1
      IF (LN-50) 470,470,460
  460 CONTINUE
      LN=1
  470 CONTINUE
      CRT(NC)=SPD
      TCRT(NC,IROT)=CRT(NC)
C*************************************C
      NC=NC+1
      LN=LN+3
      EY1(1)=0.
      EY2(1)=1.
      DTX=0.
      I=1
      IF (LN-50) 490,490,480
  480 CONTINUE
      LN=1
  490 CONTINUE
      LN=LN+2
  500 DEFL(I)=V*EY1(I)-VR1*EY2(I)
      IF (I.NE.1) GO TO 510
      EYTH(I)=V
      GO TO 520
  510 EYTH(I)=EAN1(I)*V-EAN2(I)*VR1
  520 DEFA=ABS(DEFL(I))
      DMXA=ABS(DTX)
      I=I+1
```

```
      IF (DEFA-DMXA) 540,540,530
530   DTX=DEFL(I-1)
540   IF (I-NSTAT(IROT)) 550,550,560
550   GO TO 500
560   DO 570 I=1,NSTAT(IROT)
      DPC(I)=DEFL(I)/DTX
      EYTH(I)=EYTH(I)/DTX
      EEYTH(NC-1,I)=EYTH(I)
570   DDPC(NC-1,I)=DPC(I)
      DO 600 I=1,NSTAT(IROT)
      LN=LN+1
      IF (LN-54) 590,590,580
580   CONTINUE
      LN=1
590   CONTINUE
      LN=LN+1
600   CONTINUE
      SPD=SSPD+DSP
      DETP=0.
      GO TO 290
610   CONTINUE
C************** END OF THE LOOP 290-610 ************************C
C*********************************************************************C
      DO 6200 IBEAR= 1,NBEAR(IROT)
      DO 6201 I=1,6
      DO 6201 J=1,6
      STIF2(IROT,IBEAR,I,J)= STIF_BEAR(IROT,IBEAR,I,J)
      DAMP2(IROT,IBEAR,I,J)= DAMP_BEAR(IROT,IBEAR,I,J)
 6201 CONTINUE
 6200 CONTINUE
C*********************************************************************C
C* SUBSTRACTING AVERAGE BEARING STIFFNESS FROM STIFFNESS MATRIX *C
      DO 620 IBEAR=1,NBEAR(IROT)
      STIF_AV(IROT,IBEAR,1,1)=0.5*( STIF_BEAR(IROT,IBEAR,1,1) +
     &          STIF_BEAR(IROT,IBEAR,3,3) )
      STIF_AV(IROT,IBEAR,2,2)=0.5*( STIF_BEAR(IROT,IBEAR,2,2) +
     &           STIF_BEAR(IROT,IBEAR,4,4) )
      AK= STIF_AV(IROT,IBEAR,1,1)
      AKR= STIF_AV(IROT,IBEAR,2,2)
      STIF_BEAR(IROT,IBEAR,1,1)=STIF_BEAR(IROT,IBEAR,1,1)-AK
      STIF_BEAR(IROT,IBEAR,3,3)=STIF_BEAR(IROT,IBEAR,3,3)-AK
      STIF_BEAR(IROT,IBEAR,2,2)=STIF_BEAR(IROT,IBEAR,2,2)-AKR
      STIF_BEAR(IROT,IBEAR,4,4)=STIF_BEAR(IROT,IBEAR,4,4)-AKR
  620 CONTINUE
C*********************************************************************C
C* FOR THE TRANSIENT PART OF THE PROGRAM *C

      DO 6210 IBEAR= 1,NBEAR(IROT)
      DO 6211 I=1,4
      DO 6211 J=1,4
      STIFA(IROT,IBEAR,I,J)= STIF_AV(IROT,IBEAR,I,J)
 6211 CONTINUE
 6210 CONTINUE

      DO 6220 IBEAR= 1,NBEAR(IROT)
      DO 6221 I=5,6
      DO 6221 J=5,6
      STIFA(IROT,IBEAR,I,J)= 0.0
 6221 CONTINUE
 6220 CONTINUE
C*********************************************************************C
      AK=0.0
      AKR=0.0
      DO 60 IBEAR=1,NBEAR(IROT)
      WRITE (NUNIT(NCOUNT),*) 'STIFFNESS MATRIX FOR ',IBEAR,' BEARING'
      WRITE (NUNIT(NCOUNT),2)(( STIF2(IROT,IBEAR,I,J),J=1,6),I=1,6)
```

```
      WRITE (NUNIT(NCOUNT),*)
      WRITE (NUNIT(NCOUNT),*) 'DAMPING MATRIX FOR ',IBEAR,' BEARING'
      WRITE (NUNIT(NCOUNT),2)(( DAMP2(IROT,IBEAR,I,J),J=1,6),I=1,6)
 2    FORMAT(2X, 6F12.3 )
 60   CONTINUE

C*************************************************************C
C**  NCS -> NUMBER OF CRITICAL SPEED= NSPEED II=ISPEED
      NSPEED= NC-1

C** OUTPUT THE RESULTS OF THE ABOVE CALCULATIONS ******************C
      DO 650 ISPEED=1,NSPEED
      WMOD(ISPEED)=0.
      DO 630 ISTAT=1,NSTAT(IROT)
 30   WMOD(ISPEED)=WMOD(ISPEED)+
     &          ITRANS(ISTAT)*EEYTH(ISPEED,ISTAT)**2.0 +
     &          W(ISTAT)*DDPC(ISPEED,ISTAT)**2.0
      WMOD(ISPEED)=WMOD(ISPEED)/386.4
      TWMOD(ISPEED,IROT)=WMOD(ISPEED)

      DO 640 ISTAT=1,NSTAT(IROT)
C** EEYTH - SLOPE OF THE BEAM
      EEYTH(ISPEED,ISTAT)=EEYTH(ISPEED,ISTAT)/(WMOD(ISPEED)**0.5)
C** DDPC - DEFLECTION OF THE BEAM
      DDPC(ISPEED,ISTAT)=DDPC(ISPEED,ISTAT)/(WMOD(ISPEED)**0.5)
C**  REAL LMODE(NROT,NMODE,NSTAT) - LATERAL MODE SHAPES OF THE SYSTEM
      LMODE(IROT,ISPEED,ISTAT)= DDPC(ISPEED,ISTAT)
C** TEEYTH - SLOPE OF THE BEAM
      SLOPE(IROT,ISPEED,ISTAT)=  EEYTH(ISPEED,ISTAT)
C** SHAPE ARRAY KEEPS THE RESULTS OF CALCULATIONS **************C
      SHAPE(IROT,ISPEED,ISTAT,1)= LMODE(IROT,ISPEED,ISTAT)
      SHAPE(IROT,ISPEED,ISTAT,2)= SLOPE(IROT,ISPEED,ISTAT)

640   CONTINUE
650   CONTINUE

      WRITE (NUNIT(NCOUNT-2),*) ' OUTPUT DATA '
      WRITE (NUNIT(NCOUNT-2),*) ' LATERAL FREQUENCIES AND MODE
     & SHAPES '
      WRITE (NUNIT(NCOUNT-2),*)
      WRITE (NUNIT(NCOUNT-2),*) ' ROTOR ',' #', IROT
CC      WRITE (NUNIT(NCOUNT-2),*) ' LATERAL FREQUENCIES '
      DO 660 ISPEED=1,NSPEED
      WRITE (NUNIT(NCOUNT-2),*)
      WRITE (NUNIT(NCOUNT-2),*) 'NO.',' FREQUENCY ( HZ )',
     1     ' MODAL WEIGHT '
      WRITE (NUNIT(NCOUNT-2),*) ISPEED, CRT(ISPEED)/60.0, WMOD(ISPEED)
C*************************************************************C
      FREQ(IROT,ISPEED,1)=CRT(ISPEED)
      FREQ(IROT,ISPEED,2)=CRT(ISPEED)
      WRITE (NUNIT(NCOUNT-2),*)
      WRITE (NUNIT(NCOUNT-2),*)
      WRITE (NUNIT(NCOUNT-2),*) ' MODE SHAPES FOR ROTOR NO.',IROT
      WRITE (NUNIT(NCOUNT-2),*)' #STATION    DEFLECTION         SLOPE '
      WRITE (NUNIT(NCOUNT-2),*)
      DO 660 ISTAT=1,NSTAT(IROT)
      WRITE (NUNIT(NCOUNT-2),*) ISTAT,LMODE(IROT,ISPEED,ISTAT),
     &                          SLOPE(IROT,ISPEED,ISTAT)

660   CONTINUE

      WRITE (NUNIT(NCOUNT-2),*)
      WRITE (NUNIT(NCOUNT-2),*) ' CHECK-OUT THE ORTHOGONALITY OF
     & THE MODE SHAPES '
      WRITE (NUNIT(NCOUNT-2),*) ' MODE SHAPES ARE NORMALIZED WITH
     &  RESPECT TO  MASS MATRIX '
```

```
      WRITE (NUNIT(NCOUNT-2),*)
      DO 680 ISPEED=1,NSPEED
      DO 670 JSPEED=1,NSPEED
      TMX(ISPEED,JSPEED)=0.
      DO 670 ISTAT=1,NSTAT(IROT)
      TMX(ISPEED,JSPEED)=TMX(ISPEED,JSPEED) +
     & W(ISTAT)*LMODE(IROT,ISPEED,ISTAT)*
     &          LMODE(IROT,JSPEED,ISTAT)/386.4+
     & ITRANS(ISTAT)*SLOPE(IROT,ISPEED,ISTAT)*
     &               SLOPE(IROT,JSPEED,ISTAT)/386.4
670   CONTINUE
      WRITE (NUNIT(NCOUNT-2),*) ( TMX(ISPEED,JSPEED),JSPEED=1,NSPEED )
680   CONTINUE
      NCT=NSPEED

CCCCCCCCC      NMODE(IROT)=NCT

      NM(IROT)=NCT
      WRITE (NUNIT(NCOUNT),*)
      WRITE (NUNIT(NCOUNT),*) 'NUMBER OF FOUND NATURAL FREQUENCIES'
      WRITE (NUNIT(NCOUNT),*)  NM(IROT)
      IF( NMODE(IROT) - NM(IROT) .NE. 0 ) THEN
      WRITE (NUNIT(NCOUNT),*)'THE HIGHER FREQUENCIES AND MODE SHAPES'
      WRITE (NUNIT(NCOUNT),*) ' WILL BE TAKEN AS ZEROS '
      ENDIF

C******************* TORSIONAL PART *******************************C

 **    N= NSTAT
 **    NM -> NMODE=NSPEED
C    ***************************************************************
     *  THIS PROGRAM FOR CALCULATE  Z-DIRECTION EIGEN PROBLEM  *
     ***************************************************************

C* NTYPE=1 CORRESPONDS TO TORSIONAL VIBRATION
 * NTYPE=2 CORRESPONDS TO Z-DIRECTION ( THRUST ) VIBRATION

      NTYPE=1
 700  CONTINUE
 **      NMODE=1
 **      IROT=NROT
C**      NE=IROT
      NSTATX=NSTAT(IROT)
      IF (NTYPE.EQ.1) THEN
      DO 710 ISTAT=1,NSTAT(IROT)
      GGO(ISTAT)=GGO(ISTAT)*10000000.0
      GJ(ISTAT)=32.0*DX(ISTAT)/(PI*GGO(ISTAT)*
     &          (DOUT(ISTAT)**4-DIN(ISTAT)**4))
      WJ(ISTAT)=PI*RO(ISTAT)*DX(ISTAT)*
     &          (DOUT(ISTAT)**4-DIN(ISTAT)**4)/32.0/386.4+
     &          ITRANS(ISTAT)/386.4
710   CONTINUE
      ENDIF
      IF (NTYPE.EQ.2) THEN
      DO 720 ISTAT=1,NSTAT(IROT)
      TK(ISTAT)=0.0
      EM(ISTAT)=EM(ISTAT)*1000000.0
      GJ(ISTAT)= 4.0*DX(ISTAT)/(PI*EM(ISTAT)*
     &         (DOUT(ISTAT)**2-DIN(ISTAT)**2))
      WJ(ISTAT)= PI*RO(ISTAT)*DX(ISTAT)*
     &         (DOUT(ISTAT)**2-DIN(ISTAT)**2)/4.0/386.4+
     &         EXTERNAL_WEIGHT(ISTAT)/386.4
720   CONTINUE
      ENDIF

*****************************C
```

```
      MCY=0
      ST=20.0
      AO1=0.0
  730 AO1=AO1+ST
      AO2=AO1+ST
C************************************C
      CALL SBC(NSTATX,NBC,AO1,B1,GJ,WJ,TK)
      CALL SBC(NSTATX,NBC,AO2,B2,GJ,WJ,TK)
C************************************C
      IF((B1*B2).LE.0.0) GOTO 740
      GO TO 730
  740 MCY=MCY+1
      IF(MCY.GT.NMODE(IROT)) GOTO 770
      A1=AO1
      A2=AO2
  750 IF(ABS(A1-A2).LT.0.1) GOTO 760
      AA1=A1+0.618*(A2-A1)
C**************************************C
      CALL SBC(NSTATX,NBC,AA1,BB,GJ,WJ,TK)
C**************************************C
      IF ( (B1*BB) .LE. 0.0 ) THEN
      A2=AA1
      B2=BB
      ELSE
      A1=AA1
      B1=BB
      ENDIF
      GOTO 750
  760 FR(MCY,IROT)=0.5*(A1+A2)
      IF ( NTYPE .GT. 1 ) ZF(MCY,IROT)=FR(MCY,IROT)
      FFR(MCY)=FR(MCY,IROT)/2.0/PI
      GOTO 730
  770 CONTINUE
      N1=NSTAT(IROT)+1
      DO 780 IMODE=1,NMODE(IROT)
      FV(1,IMODE)=1.0
      TV(1,IMODE)=-WJ(1)*FR(IMODE,IROT)**2
      DO 780 ISTAT=2,NSTAT(IROT)
      TV(ISTAT,IMODE)=TV(ISTAT-1,IMODE)*( 1.0+
     &     ( TK(ISTAT)-WJ(ISTAT)*FR(IMODE,IROT)**2)*GJ(ISTAT) )
     &     +FV(ISTAT-1,IMODE)*(TK(ISTAT)-WJ(ISTAT)*FR(IMODE,IROT)**2 )
      FV(ISTAT,IMODE)=TV(ISTAT-1,IMODE)*GJ(ISTAT) + FV(ISTAT-1,IMODE)
  780 CONTINUE
      DO 790 IMODE=1,NMODE(IROT)
      CO=0.0
      DO 800 ISTAT=1,NSTAT(IROT)
  800 CO=CO+FV(ISTAT,IMODE)**2*WJ(ISTAT)
      DO 790 ISTAT=1,NSTAT(IROT)
      FFV(IMODE,ISTAT,IROT)=FV(ISTAT,IMODE)/SQRT(CO)
      SHAPE(IROT,IMODE,ISTAT,5-NTYPE)=FFV(IMODE,ISTAT,IROT)
      IF (NTYPE.NE.1)  ZFV(IMODE,ISTAT,IROT)=FFV(IMODE,ISTAT,IROT)
  790 CONTINUE
      DO 810 ISTAT=1,NSTAT(IROT)
      DO 810 IMODE=1,NMODE(IROT)
  810 WF(ISTAT,IMODE)=WJ(ISTAT)*FFV(IMODE,ISTAT,IROT)
C*************************************************************C
      DO 820 IMODE=1,NMODE(IROT)
      DO 820 JMODE=1,NMODE(IROT)
      AT(IMODE,JMODE)=0.0
      DO 820 ISTAT=1,NSTAT(IROT)
  820 AT(IMODE,JMODE)=AT(IMODE,JMODE) +
     &        WF(ISTAT,JMODE)*FFV(IMODE,ISTAT,IROT)
C*************************************************************C
C********* OUTPUT OF TORSIONAL PART ***************************C
C************ OUTPUT FOR TORSIONAL VIBRATION ******************C
      IF ( NTYPE .EQ. 1 ) THEN
```

```
      WRITE ( NUNIT(NCOUNT-1),*) ' TORSIONAL MODE SHAPES  AND
     &  FREQUENCIES'
      WRITE ( NUNIT(NCOUNT-1),*)
      WRITE ( NUNIT(NCOUNT-1),1101 ) IROT
C*************** TORSIONAL  FREQUENCY ******************************C
      DO 825 IMODE=1,NMODE(IROT)
      FREQ(IROT,IMODE,5-NTYPE)=FR(IMODE,IROT)
 825  CONTINUE

      WRITE ( NUNIT(NCOUNT-1),* ) '          MODE SHAPES '
      WRITE ( NUNIT(NCOUNT-1),* ) ( IMODE, IMODE=1,NMODE(IROT) )
      WRITE ( NUNIT(NCOUNT-1),* ) ' TORSIONAL FREQUENCY ( HZ ) '
      WRITE ( NUNIT(NCOUNT-1),* ) ( FREQ(IROT,IMODE,4)/60.0,
     &                             IMODE=1,NMODE(IROT) )
      DO 830 ISTAT=1,NSTAT(IROT)
      WRITE ( NUNIT(NCOUNT-1),* ) ISTAT,(SHAPE(IROT,IMODE,ISTAT,4),
     &                           IMODE=1,NMODE(IROT))
 830  CONTINUE
      ENDIF
C************* OUTPUT FOR Z-DIRECTION VIBRATION ******************************C
      IF ( NTYPE .EQ. 2 ) THEN
      WRITE ( NUNIT(NCOUNT-1),*) '    Z-DIRECTION MODE SHAPES '
      WRITE ( NUNIT(NCOUNT-1),2101 ) IROT
C*************** Z-DIRECTION  FREQUENCY ******************************C
      DO 835 IMODE=1,NMODE(IROT)
      FREQ(IROT,IMODE,5-NTYPE)=ZF(IMODE,IROT)
 835  CONTINUE

      WRITE ( NUNIT(NCOUNT-1),* ) ' MODE SHAPES '
      WRITE ( NUNIT(NCOUNT-1),* ) ( IMODE, IMODE=1,NMODE(IROT) )
      WRITE ( NUNIT(NCOUNT-1),* ) '    Z-DIRECTION FREQUENCY ( HZ ) '
      WRITE ( NUNIT(NCOUNT-1),* ) '      ', (  FREQ(IROT,IMODE,3)/60.0,
     &                              IMODE=1,NMODE(IROT) )
      WRITE ( NUNIT(NCOUNT-1),* )
      DO 840 ISTAT=1,NSTAT(IROT)
      WRITE ( NUNIT(NCOUNT-1),* ) ISTAT,(SHAPE(IROT,IMODE,ISTAT,3),
     &                           IMODE=1,NMODE(IROT))
 840  CONTINUE
      ENDIF

C*****************************************************************C

 61   FORMAT (I2, 8F8.3 )
1101  FORMAT('TORSIONAL FREQUENCY AND ORTHONOMAL MODE ** IROT=',I2)
1102  FORMAT(3X,'IMODE=',I3,30X,'FREQUENCY=',2F10.3)
1103  FORMAT(2X,' ISTAT=',14X,'FV(J,I)=',12X,'FFV(J,I)',8X,'TV=')
1104  FORMAT(2X,I3,8X,G14.6,8X,F12.6,6X,G12.5)
1105  FORMAT (3(/2X,3F16.8))
1109  FORMAT (2X,I5,2F13.5)
2101  FORMAT('Z-DIRECTION FREQUENCY AND ORTHONOMAL MODE * IROT=',I2)
2102  FORMAT(3X,'IMODE=',I3,30X,'FREQUENCY=',F10.3)
2103  FORMAT(2X,' ISTAT=',14X,'ZV(J,I)=',12X,'ZFV(J,I)',8X,'FZ=')
2109  FORMAT (2X,I5,F13.5,2E15.8)

C** CARD FOR THE END OF THE TORSIONAL AND Z-DIRECTION PART **C
      NTYPE= NTYPE + 1
      IF ( NTYPE-1 .EQ. 1 ) GOTO 700

C****************** CARD FOR THE END OF THE SUBROUTINE ***********C
C*****************************************************************C
C* CHECK IF THERE ARE MORE ROTORS TO CALCULATE THE MODE SHAPES ****C
      IROT=IROT+1
      IF ( IROT .LE. NROT ) GOTO 10
C*****************************************************************C
C*****************************************************************C
C******** INITIALIZATION NCON_BEAR TABLE *****************************C
```

```
      DO 201 IROT=1,NROT
      DO 201 IBEAR=1,NBEAR(IROT)
      DO 201 JROT=1,NROT
      DO 201 JBEAR=1,NBEAR(JROT)
      NCON_BEAR(IROT,IBEAR,JROT,JBEAR)=0
  201 CONTINUE

C*****************************************C
C* INPUT FOR BEAR CONNECTIONS ************C
C*****************************************C
      READ (NUNIT(1),*)
      READ (NUNIT(1),*) JCHECK4
C* JCHECK4 IS AN INTEGER SHOWING THE NUMBER OF LINES IN A BEARING CONNECTION TAB
LE.
C* THE BEARING CONNECTION TABLE IS IN THE FORM:
C* IROT IBEAR JROT JBEAR ITYPE_CON
C* IROT      - I-TH ROTOR INDEX
C* IBEAR     - I-TH BEARING OF I-TH ROTOR INDEX
C* JROT      - J-TH ROTOR INDEX
C* JBEAR     - J-TH BEARING OF J-TH ROTOR INDEX
C* ITYPE_CON - TYPE OF CONNECTION BETWEEN (IROT,IBEAR) AND (JROTOR,JBEAR)
C* IF ITYPE_CON =0 THEN BEARING IS CONNECTED TO THE GROUND,AND
C* THERE IS NO CONNECTION TO ANOTHER BEARING OR TO CASING.
C* IF ITYPE_CON !=0 IT SHOWS THE TYPE OF BEARING STIFFNESS USED
C* IN THE CALCULATION OF THE BEARING FORCE BETWEEN (IROT,IBEAR) AND (JROTOR,JBEA
R).
C* THE TYPE OF BEARING STIFFNESS IS THE NUMBER OF THE BEARING STIFFNESS TABLE.

      WRITE (NUNIT(NCOUNT),*)
      WRITE (NUNIT(NCOUNT),*) ' BEARINGS CONNECTION TABLE '
      WRITE (NUNIT(NCOUNT),*) ' IROT  IBEAR  JROT  JBEAR
     &                   ITYPE_CON '

      IF( JCHECK4 .EQ. 0 ) THEN
      WRITE (NUNIT(NCOUNT),*)' ALL BEARINGS ARE CONNECTED TO
     & GROUND'
      ENDIF

      DO 203 JJ=1,JCHECK4
C* IROT,IBEAR CONNECTED TO JROT,JBEAR, TYPE OF CONNECTION
      READ (NUNIT(1),*) IROT,IBEAR,JROT,JBEAR,
     &                  ITYPE_CON
      NCON_BEAR(IROT,IBEAR,JROT,JBEAR)=ITYPE_CON
      NCON_BEAR(JROT,JBEAR,IROT,IBEAR)=ITYPE_CON

      WRITE (NUNIT(NCOUNT),*) IROT,' ',IBEAR,' ',JROT,' ',JBEAR,' ',
     &                        ITYPE_CON
      WRITE (NUNIT(NCOUNT),*)
  203 CONTINUE

      IF( NSWITCH(1) .EQ. 0 ) GOTO 1000

C*****************************************************************C
C************************ END OF THE MAIN BODY OF LAT ***************C
C*****************************************************************C
C* WE JUST FOUND ABOVE THE PLANAR UNDAMPED NATURAL FREQUENCIES AND   *C
C* MODE SHAPES.NOW WE PROCEED FOR THE CALCULATIONS OF THE DAMPED     *C
C* MODE SHAPES AND FREQUENCIES WHICH ARE NEEDED FOR THE STABILITY    *C
C* ANALYSIS OF THE PROBLEM.HERE WE JUST CALCULATE A GLOBAL MATRIX    *C
C* FOR THE EIGENVALUE PROBLEM.USING THIS GLOBAL MATRIX YOU CAN FIND  *C
C* THE EIGENVALUES AND MODE SHAPES BY ANY OTHER STANDART PROGRAM.    *C
C* THIS PART IS ADDED TO CREATE A GLOBAL MATRIX FOR THE EIGENVALUE   *C
C* SOLVER PROGRAM.                                                  *C
C*****************************************************************C
      DO 950 IROT=1,NROT
      NSPEED= NMODE(IROT)
```

```
DO 911 I=1,NSPEED
```

```
      DO 911 J=1,NSPEED
      KBAR(I,J)=0.0
      CBAR(I,J)=0.0
911   CONTINUE

      DO 900 IBEAR=1,NBEAR(IROT)

      ISTAT= LBEAR(IROT,IBEAR)
      DO 901 ISPEED=1,NSPEED
      FMODE(IROT,ISPEED,1)= SHAPE(IROT,ISPEED,ISTAT,1)
      FMODE(IROT,ISPEED,2)= SHAPE(IROT,ISPEED,ISTAT,2)
      FMODE(IROT,ISPEED,3)= FMODE(IROT,ISPEED,1)
      FMODE(IROT,ISPEED,4)= FMODE(IROT,ISPEED,2)
901   CONTINUE

      DO 910 ISPEED=1,NSPEED
      DO 910 JSPEED=1,NSPEED
      SBARK= 0.0
      SBARC= 0.0
      DO 902 I=1,2
      DO 902 J=1,2
      SBARK= SBARK +
     & FMODE(IROT,ISPEED,I)*STIF_BEAR(IROT,IBEAR,I,J)
     & *FMODE(IROT,JSPEED,J)
      SBARC= SBARC +
     & FMODE(IROT,ISPEED,I)*DAMP_BEAR(IROT,IBEAR,I,J)
     & *FMODE(IROT,JSPEED,J)
902   CONTINUE
      KBAR(ISPEED,JSPEED)= KBAR(ISPEED,JSPEED) + SBARK
      CBAR(ISPEED,JSPEED)= CBAR(ISPEED,JSPEED) + SBARC
910   CONTINUE
900   CONTINUE

C       WRITE(56,* ) ((CBAR(I,J),J=1,NSPEED),I=1,NSPEED)

      DO 920 I=1,2*NSPEED
      DO 920 J=1,2*NSPEED
      AGLOB(IROT,I,J)= 0.0
920   CONTINUE
C* FORM IDENTITY MATRIX BLOCK *C
      DO 925 I=1,NSPEED
      DO 925 J=NSPEED+1,2*NSPEED
      IF( I .EQ. J-NSPEED ) AGLOB(IROT,I,J)= 1.0
925   CONTINUE
      DO 930 I=NSPEED+1, 2*NSPEED
      DO 930 J=1,NSPEED
      AGLOB(IROT,I,J)= -KBAR(I-NSPEED,J)
      IF( J .EQ. I-NSPEED )
     &  AGLOB(IROT,I,J)= AGLOB(IROT,I,J)-
     &  (FREQ(IROT,J,1)/9.554)**2.0
930   CONTINUE
      DO 935 I=NSPEED+1,2*NSPEED
      DO 935 J=NSPEED+1,2*NSPEED
      AGLOB(IROT,I,J)= -CBAR(I-NSPEED,J-NSPEED)
935   CONTINUE

      WRITE(30+IROT,*)' MATRIX 2*NSPEEDx2*NSPEED '
      WRITE(30+IROT,*) 2*NSPEED, 2*NSPEED
      WRITE(30+IROT,*)' GLOBAL MATRIX FOR IROT ',IROT

      WRITE(30+IROT,940)
     &      ((AGLOB(IROT,I,J),J=1,2*NSPEED),I=1,2*NSPEED)
940   FORMAT( 6(G11.3,1X) )

950   CONTINUE
```

```
C*****************************************************************C
C*******  START CALCULATIONS SUPER-GLOBAL MATRIX **************C
C*****************************************************************C
      DO 921 I=1,2*NSPEED*NROT
      DO 921 J=1,2*NSPEED*NROT
      ASUPER(I,J)= 0.0
 921  CONTINUE

      DO 952 IROT=1,NROT
      DO 952 I=1,2*NSPEED
      DO 952 J=1,2*NSPEED
      II= (IROT-1)*2*NSPEED + I
      JJ= (IROT-1)*2*NSPEED + J
      ASUPER(II,JJ)= AGLOB(IROT,I,J)
 952  CONTINUE


      DO 955 IROT=1,NROT
      DO 956 JROT=1,NROT
      IF( IROT .EQ. JROT ) GOTO 956

C*********************************C
      IF( NMODE(IROT) - NMODE(JROT) .NE. 0 ) THEN
      WRITE(NUNIT(NCOUNT), *) 'WARNING:', IROT ,'AND ',JROT,
     & ' HAVE DIFFERENT NUMBER OF MODES '
      WRITE(NUNIT(NCOUNT), *) 'A-SUPER WILL NOT BE CALCULATED '
      WRITE(NUNIT(NCOUNT), *) 'CHANGE NUMBER OF MODE SHAPES '
      STOP
      ENDIF
C*********************************C

      DO 958 I=1,NSPEED
      DO 958 J=1,NSPEED
      K2BAR(I,J)=0.0
      C2BAR(I,J)=0.0
 958  CONTINUE

      DO 957 IBEAR= 1,NBEAR(IROT)
      DO 959 JBEAR= 1,NBEAR(JROT)
      IF( NCON_BEAR(IROT,IBEAR,JROT,JBEAR) .NE. 1 ) GOTO 959
      NSPEED= NMODE(IROT)
      ISTAT= LBEAR(IROT,IBEAR)
      JSTAT= LBEAR(JROT,JBEAR)
      DO 960 ISPEED=1,NSPEED
      FMODE(IROT,ISPEED,1)= SHAPE(IROT,ISPEED,ISTAT,1)
      FMODE(IROT,ISPEED,2)= SHAPE(IROT,ISPEED,ISTAT,2)
      FMODE(IROT,ISPEED,3)= FMODE(IROT,ISPEED,1)
      FMODE(IROT,ISPEED,4)= FMODE(IROT,ISPEED,2)
      FMODE(JROT,ISPEED,1)= SHAPE(JROT,ISPEED,JSTAT,1)
      FMODE(JROT,ISPEED,2)= SHAPE(JROT,ISPEED,JSTAT,2)
      FMODE(JROT,ISPEED,3)= FMODE(JROT,ISPEED,1)
      FMODE(JROT,ISPEED,4)= FMODE(JROT,ISPEED,2)
 960  CONTINUE

      DO 961 ISPEED=1,NSPEED
      DO 961 JSPEED=1,NSPEED
      DO 962 I=1,2
      SBARK= 0.0
      SBARC= 0.0
      DO 963 J=1,2
      SBARK= SBARK +
     & ( STIF_BEAR(IROT,IBEAR,I,J)+STIF_AV(IROT,IBEAR,I,J) )
     &  *FMODE(JROT,JSPEED,J)
      SBARC= SBARC + DAMP_BEAR(IROT,IBEAR,I,J)*FMODE(JROT,JSPEED,J)
 963  CONTINUE
      K2BAR(ISPEED,JSPEED)= K2BAR(ISPEED,JSPEED) +
```

```
     & FMODE(IROT,ISPEED,I)*SBARK
      C2BAR(ISPEED,JSPEED)= C2BAR(ISPEED,JSPEED) +
     & FMODE(IROT,ISPEED,I)*SBARC
 962  CONTINUE
 961  CONTINUE


 959  CONTINUE
 957  CONTINUE

C****C
      DO 964 I=IROT*2*NSPEED, IROT*2*NSPEED+NSPEED-1
      DO 964 J=JROT*2*NSPEED, JROT*2*NSPEED+NSPEED-1
      ASUPER(I-NSPEED+1,J-NSPEED+1)=
     &              C2BAR(I-IROT*2*NSPEED+1,J-JROT*2*NSPEED+1)
      ASUPER(I-NSPEED+1,J-2*NSPEED+1)=
     &              K2BAR(I-IROT*2*NSPEED+1,J-JROT*2*NSPEED+1)

 964  CONTINUE

 956  CONTINUE
 955  CONTINUE

      WRITE(40,*)' MATRIX 2*NROT*NSPEEDx2*NROT*NSPEED '
      WRITE(40,*) 2*NROT*NSPEED, 2*NROT*NSPEED
      WRITE(40,*)' GLOBAL MATRIX FOR NROT ',NROT

      WRITE(40,967)
     & ((ASUPER(I,J),J=1,2*NROT*NSPEED),I=1,2*NROT*NSPEED)
 967  FORMAT( 6(G10.3,1X) )


1000  CONTINUE
      RETURN
      END

      SUBROUTINE SBC(NSTATX,NBC,OM,A,GJ,WJ,TK)
C     ****************************************************************
C     *  THIS IS TO SET BOUNDARY CONDITIONS FOR MATRIX TRANSFORM METHOD  *
C     ****************************************************************
      INTEGER NBC
      REAL D(2,2),B(2,2),C(2,2)
      REAL GJ(NSTATX),WJ(NSTATX), TK(NSTATX)
      C(2,2)=1.0
      B(1,1)=1.0
      B(1,2)=0.0
      B(2,1)=0.0
      B(2,2)=1.0
      DO 20 I=1,NSTATX
      C(1,1)=1.0+(TK(I)-WJ(I)*OM**2)*GJ(I)
      C(1,2)=TK(I) -WJ(I)*OM**2
      C(2,1)=1.0*GJ(I)
      D(1,1)=C(1,1)*B(1,1)+C(1,2)*B(2,1)
      D(1,2)=C(1,1)*B(1,2)+C(1,2)*B(2,2)
      D(2,1)=C(2,1)*B(1,1)+C(2,2)*B(2,1)
      D(2,2)=C(2,1)*B(1,2)+C(2,2)*B(2,2)
      B(1,1)=D(1,1)
      B(1,2)=D(1,2)
      B(2,1)=D(2,1)
      B(2,2)=D(2,2)
 20   CONTINUE
      NBC=2
      FREE---------FIXED -> NBC=1
      IF( NBC .EQ. 1 ) A=B(2,2)
C     FREE---------FREE -> NBC=2
      IF( NBC .EQ. 2 ) A=B(1,2)
```

```
      FIXED--------FIXED -> NBC=3
      IF( NBC .EQ. 3 ) A=B(2,1)
      FIXED--------FREE -> NBC=4
      IF( NBC .EQ. 4 ) A=B(1,1)
      RETURN
      END

C**********************************************************************C
```